



## Interface and Hardware Commands

---

- [debug fastethernet](#), on page 3
- [debug ilpower](#), on page 4
- [debug interface](#), on page 5
- [debug lldp packets](#), on page 6
- [debug nmsp](#), on page 7
- [duplex](#), on page 8
- [errdisable detect cause](#), on page 10
- [errdisable detect cause small-frame](#), on page 12
- [errdisable recovery cause](#), on page 13
- [errdisable recovery cause small-frame](#), on page 16
- [errdisable recovery interval](#), on page 17
- [lldp \(interface configuration\)](#), on page 18
- [mdix auto](#), on page 19
- [network-policy](#), on page 20
- [network-policy profile \(global configuration\)](#), on page 21
- [nmsp attachment suppress](#), on page 22
- [power efficient-ethernet auto](#), on page 23
- [power inline](#), on page 24
- [power inline consumption](#), on page 27
- [power inline police](#), on page 30
- [show eee](#), on page 32
- [show env](#), on page 35
- [show errdisable detect](#), on page 37
- [show errdisable recovery](#), on page 38
- [show interfaces](#), on page 39
- [show interfaces counters](#), on page 44
- [show interfaces switchport](#), on page 46
- [show interfaces transceiver](#), on page 50
- [show ip ports all](#), on page 53
- [show network-policy profile](#), on page 54
- [show power inline](#), on page 55
- [show system mtu](#), on page 60
- [speed](#), on page 61

- [switchport backup interface](#), on page 63
- [switchport block](#), on page 65
- [system mtu](#), on page 66
- [voice-signaling vlan \(network-policy configuration\)](#), on page 67
- [voice vlan \(network-policy configuration\)](#), on page 69

# debug fastethernet

To enable debugging of the Ethernet management port, use the **debug fastethernet** command in EXEC mode. To disable debugging, use the **no** form of this command.

```
debug fastethernet {af | events | packets}
no debug fastethernet {af | events | packets}
```

## Syntax Description

<b>af</b>	Displays Ethernet management port software-address-filter debug messages.
<b>events</b>	Displays Ethernet management port event debug messages.
<b>packets</b>	Displays Ethernet management port packet debug messages.

## Command Default

Debugging is disabled.

## Command Modes

User EXEC  
Privileged EXEC

## Command History

Release	Modification
Cisco IOS Release 15.0(2)EX	This command was introduced.

## Usage Guidelines

The **undebug fastethernet { af | events | packets}** command is the same as the **no debug fastethernet {af | events | packets}** command.

When you enable debugging on a switch stack, it is enabled only on the active switch. To enable debugging on a member switch, you can start a session from the active switch by using the **session switch-number EXEC** command. Then enter the **debug** command at the command-line prompt of the member switch. You also can use the **remote command stack-member-number LINE EXEC** command on the active switch to enable debugging on a member switch without first starting a session.

## Related Commands

Command	Description
show debugging	Displays information about the types of debugging that are enabled.

# debug ilpower

To enable debugging of the power controller and Power over Ethernet (PoE) system, use the **debug ilpower** command in privileged EXEC mode. To disable debugging, use the **no** form of this command.

```
debug ilpower {cdp | controller | event | ha | port | powerman | registries | scp | sense}
no debug ilpower {cdp | controller | event | ha | port | powerman | registries | scp | sense}
```

## Syntax Description

<b>cdp</b>	Displays PoE Cisco Discovery Protocol (CDP) debug messages.
<b>controller</b>	Displays PoE controller debug messages.
<b>event</b>	Displays PoE event debug messages.
<b>ha</b>	Displays PoE high-availability messages.
<b>port</b>	Displays PoE port manager debug messages.
<b>powerman</b>	Displays PoE power management debug messages.
<b>registries</b>	Displays PoE registries debug messages.
<b>scp</b>	Displays PoE SCP debug messages.
<b>sense</b>	Displays PoE sense debug messages.

## Command Default

Debugging is disabled.

## Command Modes

Privileged EXEC

## Command History

Release	Modification
Cisco IOS Release 15.0(2)EX	This command was introduced.

## Usage Guidelines

This command is supported only on PoE-capable switches.

When you enable debugging on a switch stack, it is enabled only on the active switch. To enable debugging on a member switch, you can start a session from the active switch by using the **session** *switch-number* EXEC command. Then enter the **debug** command at the command-line prompt of the member switch. You also can use the **remote command** *stack-member-number* *LINE* EXEC command on the active switch to enable debugging on a member switch without first starting a session.

# debug interface

To enable debugging of interface-related activities, use the **debug interface** command in privileged EXEC mode. To disable debugging, use the **no** form of this command.

```
debug interface {interface-id | counters {exceptions | protocol memory} | null interface-number |
port-channel port-channel-number | states | vlan vlan-id}
no debug interface {interface-id | counters {exceptions | protocol memory} | null interface-number |
port-channel port-channel-number | states | vlan vlan-id}
```

## Syntax Description

<i>interface-id</i>	ID of the physical interface. Displays debug messages for the specified physical port, identified by type switch number/module number/port, for example, gigabitethernet 1/0/2.
<b>null</b> <i>interface-number</i>	Displays debug messages for null interfaces. The interface number is always <b>0</b> .
<b>port-channel</b> <i>port-channel-number</i>	Displays debug messages for the specified EtherChannel port-channel interface. The <i>port-channel-number</i> range is 1 to 48.
<b>vlan</b> <i>vlan-id</i>	Displays debug messages for the specified VLAN. The vlan range is 1 to 4094.
<b>counters</b>	Displays counters debugging information.
<b>exceptions</b>	Displays debug messages when a recoverable exceptional condition occurs during the computation of the interface packet and data rate statistics.
<b>protocol memory</b>	Displays debug messages for memory operations of protocol counters.
<b>states</b>	Displays intermediary debug messages when an interface's state transitions.

## Command Default

Debugging is disabled.

## Command Modes

Privileged EXEC

## Command History

Release	Modification
Cisco IOS Release 15.0(2)EX	This command was introduced.

## Usage Guidelines

If you do not specify a keyword, all debug messages appear.

The **undebug interface** command is the same as the **no debug interface** command.

When you enable debugging on a switch stack, it is enabled only on the active switch. To enable debugging on a member switch, you can start a session from the active switch by using the **session** *switch-number* EXEC command. Then enter the **debug** command at the command-line prompt of the member switch. You also can use the **remote command** *stack-member-number* *LINE* EXEC command on the active switch to enable debugging on a member switch without first starting a session.

## debug lldp packets

To enable debugging of Link Layer Discovery Protocol (LLDP) packets, use the **debug lldp packets** command in privileged EXEC mode. To disable debugging, use the **no** form of this command.

**debug lldp packets**  
**no debug lldp packets**

**Syntax Description** This command has no arguments or keywords.

**Command Default** Debugging is disabled.

**Command Modes** Privileged EXEC

Command History	Release	Modification
	Cisco IOS Release 15.0(2)EX	This command was introduced.

**Usage Guidelines** The **undebug lldp packets** command is the same as the **no debug lldp packets** command. When you enable debugging on a switch stack, it is enabled only on the . To enable debugging on a member switch, you can start a session from the by using the **session *switch-number*** EXEC command.

## debug nmsp

To enable debugging of the Network Mobility Services Protocol (NMSP) on the switch, use the **debug nmsp** command in privileged EXEC mode. To disable debugging, use the **no** form of this command.

Syntax Description	all	Displays all NMSP debug messages.
	<b>connection</b>	Displays debug messages for NMSP connection events.
	<b>error</b>	Displays debugging information for NMSP error messages.
	<b>event</b>	Displays debug messages for NMSP events.
	<b>rx</b>	Displays debugging information for NMSP receive messages.
	<b>tx</b>	Displays debugging information for NMSP transmit messages.
	<b>packet</b>	Displays debug messages for NMSP packet events.

**Command Default** Debugging is disabled.

**Command Modes** Privileged EXEC

Command History	Release	Modification
	Cisco IOS Release 15.0(2)EX	This command was introduced.

### Usage Guidelines



**Note** Attachment information is not supported in Cisco IOS XE Denali 16.1.1 and later releases.

The **undebbug nmsp** command is the same as the **no debug nmsp** command.

When you enable debugging on a switch stack, it is enabled only on the active switch. To enable debugging on a member switch, you can start a session from the active switch by using the **session switch-number EXEC** command. Then enter the **debug** command at the command-line prompt of the member switch. You also can use the **remote command stack-member-number LINE EXEC** command on the active switch to enable debugging on a member switch without first starting a session.

# duplex

To specify the duplex mode of operation for a port, use the **duplex** command in interface configuration mode. To return to the default value, use the **no** form of this command.

**duplex** {**auto** | **full** | **half**}  
**no duplex** {**auto** | **full** | **half**}

## Syntax Description

**auto** Enables automatic duplex configuration. The port automatically detects whether it should run in full- or half-duplex mode, depending on the attached device mode.

**full** Enables full-duplex mode.

**half** Enables half-duplex mode (only for interfaces operating at 10 or 100 Mb/s). You cannot configure half-duplex mode for interfaces operating at 1000 or 10,000 Mb/s.

## Command Default

The default is **auto** for Fast Ethernet and Gigabit Ethernet ports.

The default is **half** for 100BASE-*x* (where *x* is -BX, -FX, -FX-FE, or -LX) SFP modules.

Duplex options are not supported on the 1000BASE-*x* or 10GBASE-*x* (where *x* is -BX, -CWDM, -LX, -SX, or -ZX) small form-factor pluggable (SFP) modules.

## Command Modes

Interface configuration

## Command History

Release	Modification
Cisco IOS Release 15.0(2)EX	This command was introduced.

## Usage Guidelines

For Fast Ethernet ports, setting the port to **auto** has the same effect as specifying **half** if the attached device does not autonegotiate the duplex parameter.

For Gigabit Ethernet ports, setting the port to **auto** has the same effect as specifying **full** if the attached device does not autonegotiate the duplex parameter.



**Note** Half-duplex mode is supported on Gigabit Ethernet interfaces if the duplex mode is **auto** and the connected device is operating at half duplex. However, you cannot configure these interfaces to operate in half-duplex mode.

Certain ports can be configured to be either full duplex or half duplex. How this command is applied depends on the device to which the switch is attached.

If both ends of the line support autonegotiation, we highly recommend using the default autonegotiation settings. If one interface supports autonegotiation and the other end does not, configure duplex and speed on both interfaces, and use the **auto** setting on the supported side.

If the speed is set to **auto**, the switch negotiates with the device at the other end of the link for the speed setting and then forces the speed setting to the negotiated value. The duplex setting remains as configured on each end of the link, which could result in a duplex setting mismatch.



You can configure the duplex setting when the speed is set to **auto**.



---

**Caution** Changing the interface speed and duplex mode configuration might shut down and reenble the interface during the reconfiguration.

---

You can verify your setting by entering the **show interfaces** privileged EXEC command.

---

## Examples

This example shows how to configure an interface for full-duplex operation:

```
Device(config)# interface gigabitethernet1/0/1
Device(config-if)# duplex full
```

## errdisable detect cause

To enable error-disable detection for a specific cause or for all causes, use the **errdisable detect cause** command in global configuration mode. To disable the error-disable detection feature, use the **no** form of this command.

```
errdisable detect cause {all | arp-inspection | bpduguard shutdown vlan | dhcp-rate-limit | dtp-flap
| gbic-invalid | inline-power | link-flap | loopback | pagp-flap | pppoe-ia-rate-limit | psp shutdown
vlan | security-violation shutdown vlan | sfp-config-mismatch}
no errdisable detect cause {all | arp-inspection | bpduguard shutdown vlan | dhcp-rate-limit | dtp-flap
| gbic-invalid | inline-power | link-flap | loopback | pagp-flap | pppoe-ia-rate-limit | psp shutdown
vlan | security-violation shutdown vlan | sfp-config-mismatch}
```

### Syntax Description

<b>all</b>	Enables error detection for all error-disabled causes.
<b>arp-inspection</b>	Enables error detection for dynamic Address Resolution Protocol (ARP) inspection.
<b>bpduguard shutdown vlan</b>	Enables per-VLAN error-disable for BPDU guard.
<b>dhcp-rate-limit</b>	Enables error detection for DHCP snooping.
<b>dtp-flap</b>	Enables error detection for the Dynamic Trunking Protocol (DTP) flapping.
<b>gbic-invalid</b>	Enables error detection for an invalid Gigabit Interface Converter (GBIC) module.  <b>Note</b> This error refers to an invalid small form-factor pluggable (SFP) module.
<b>inline-power</b>	Enables error detection for the Power over Ethernet (PoE) error-disabled cause.  <b>Note</b> This keyword is supported only on switches with PoE ports.
<b>link-flap</b>	Enables error detection for link-state flapping.
<b>loopback</b>	Enables error detection for detected loopbacks.
<b>pagp-flap</b>	Enables error detection for the Port Aggregation Protocol (PAgP) flap error-disabled cause.
<b>pppoe-ia-rate-limit</b>	Enables error detection for the PPPoE Intermediate Agent rate-limit error-disabled cause.
<b>psp shutdown vlan</b>	Enables error detection for protocol storm protection (PSP).
<b>security-violation shutdown vlan</b>	Enables voice aware 802.1x security.
<b>sfp-config-mismatch</b>	Enables error detection on an SFP configuration mismatch.

**Command Default** Detection is enabled for all causes. All causes, except per-VLAN error disabling, are configured to shut down the entire port.

**Command Modes** Global configuration

<b>Command History</b>	<b>Release</b>	<b>Modification</b>
	Cisco IOS Release 15.0(2)EX	This command was introduced.

**Usage Guidelines** A cause (such as a link-flap or dhcp-rate-limit) is the reason for the error-disabled state. When a cause is detected on an interface, the interface is placed in an error-disabled state, an operational state that is similar to a link-down state.

When a port is error-disabled, it is effectively shut down, and no traffic is sent or received on the port. For the bridge protocol data unit (BPDU) guard, voice-aware 802.1x security, and port-security features, you can configure the switch to shut down only the offending VLAN on the port when a violation occurs, instead of shutting down the entire port.

If you set a recovery mechanism for the cause by entering the **errdisable recovery** global configuration command, the interface is brought out of the error-disabled state and allowed to retry the operation when all causes have timed out. If you do not set a recovery mechanism, you must enter the **shutdown** and then the **no shutdown** commands to manually recover an interface from the error-disabled state.

For protocol storm protection, excess packets are dropped for a maximum of two virtual ports. Virtual port error disabling using the **psp** keyword is not supported for EtherChannel and Flexlink interfaces.

To verify your settings, enter the **show errdisable detect** privileged EXEC command.

This example shows how to enable error-disabled detection for the link-flap error-disabled cause:

```
Device(config)# errdisable detect cause link-flap
```

This command shows how to globally configure BPDU guard for a per-VLAN error-disabled state:

```
Device(config)# errdisable detect cause bpduguard shutdown vlan
```

This command shows how to globally configure voice-aware 802.1x security for a per-VLAN error-disabled state:

```
Device(config)# errdisable detect cause security-violation shutdown vlan
```

You can verify your setting by entering the **show errdisable detect** privileged EXEC command.

## errdisable detect cause small-frame

To allow any switch port to be error disabled if incoming VLAN-tagged packets are small frames (67 bytes or less) and arrive at the minimum configured rate (the threshold), use the **errdisable detect cause small-frame** global configuration command on the switch stack or on a standalone switch. Use the **no** form of this command to return to the default setting.

**errdisable detect cause small-frame**  
**no errdisable detect cause small-frame**

**Syntax Description** This command has no arguments or keywords.

**Command Default** This feature is disabled.

**Command Modes** Global configuration

Command History	Release	Modification
	Cisco IOS Release 15.0(2)EX	This command was introduced.

**Usage Guidelines** This command globally enables the small-frame arrival feature. Use the **small violation-rate** interface configuration command to set the threshold for each port.

You can configure the port to be automatically re-enabled by using the **errdisable recovery cause small-frame** global configuration command. You configure the recovery time by using the **errdisable recovery interval interval** global configuration command.

### Examples

This example shows how to enable the switch ports to be put into the error-disabled mode if incoming small frames arrive at the configured threshold:

```
Device(config)# errdisable detect cause small-frame
```

You can verify your setting by entering the **show interfaces** privileged EXEC command.

## errdisable recovery cause

To enable the error-disabled mechanism to recover from a specific cause, use the **errdisable recovery cause** command in global configuration mode. To return to the default setting, use the **no** form of this command.

```
errdisable recovery cause {all | arp-inspection | bpduguard | channel-misconfig | dhcp-rate-limit |
dtp-flap | gbic-invalid | inline-power | link-flap | loopback | mac-limit | pagp-flap | port-mode-failure |
pppoe-ia-rate-limit | psecure-violation | psp | security-violation | sfp-config-mismatch | storm-control |
udld | vmpls}
```

```
no errdisable recovery cause {all | arp-inspection | bpduguard | channel-misconfig | dhcp-rate-limit |
dtp-flap | gbic-invalid | inline-power | link-flap | loopback | mac-limit | pagp-flap | port-mode-failure |
pppoe-ia-rate-limit | psecure-violation | psp | security-violation | sfp-config-mismatch | storm-control |
udld | vmpls}
```

### Syntax Description

<b>all</b>	Enables the timer to recover from all error-disabled causes.
<b>arp-inspection</b>	Enables the timer to recover from the Address Resolution Protocol (ARP) inspection error-disabled state.
<b>bpduguard</b>	Enables the timer to recover from the bridge protocol data unit (BPDU) guard error-disabled state.
<b>channel-misconfig</b>	Enables the timer to recover from the EtherChannel misconfiguration error-disabled state.
<b>dhcp-rate-limit</b>	Enables the timer to recover from the DHCP snooping error-disabled state.
<b>dtp-flap</b>	Enables the timer to recover from the Dynamic Trunking Protocol (DTP) flap error-disabled state.
<b>gbic-invalid</b>	Enables the timer to recover from an invalid Gigabit Interface Converter (GBIC) module error-disabled state.  <b>Note</b> This error refers to an invalid small form-factor pluggable (SFP) error-disabled state.
<b>inline-power</b>	Enables the timer to recover from the Power over Ethernet (PoE) error-disabled state.  This keyword is supported only on switches with PoE ports.
<b>link-flap</b>	Enables the timer to recover from the link-flap error-disabled state.
<b>loopback</b>	Enables the timer to recover from a loopback error-disabled state.
<b>mac-limit</b>	Enables the timer to recover from the mac limit error-disabled state.
<b>pagp-flap</b>	Enables the timer to recover from the Port Aggregation Protocol (PAgP)-flap error-disabled state.

<b>port-mode-failure</b>	Enables the timer to recover from the port mode change failure error-disabled state.
<b>pppoe-ia-rate-limit</b>	Enables the timer to recover from the PPPoE IA rate limit error-disabled state.
<b>psecure-violation</b>	Enables the timer to recover from a port security violation disable state.
<b>psp</b>	Enables the timer to recover from the protocol storm protection (PSP) error-disabled state.
<b>security-violation</b>	Enables the timer to recover from an IEEE 802.1x-violation disabled state.
<b>sfp-config-mismatch</b>	Enables error detection on an SFP configuration mismatch.
<b>storm-control</b>	Enables the timer to recover from a storm control error.
<b>udld</b>	Enables the timer to recover from the UniDirectional Link Detection (UDLD) error-disabled state.
<b>vmmps</b>	Enables the timer to recover from the VLAN Membership Policy Server (VMPS) error-disabled state.

**Command Default** Recovery is disabled for all causes.

**Command Modes** Global configuration

<b>Command History</b>	<b>Release</b>	<b>Modification</b>
	Cisco IOS Release 15.0(2)EX	This command was introduced.

**Usage Guidelines** A cause (such as all or BPDU guard) is defined as the reason that the error-disabled state occurred. When a cause is detected on an interface, the interface is placed in the error-disabled state, an operational state similar to link-down state.

When a port is error-disabled, it is effectively shut down, and no traffic is sent or received on the port. For the BPDU guard and port-security features, you can configure the switch to shut down only the offending VLAN on the port when a violation occurs, instead of shutting down the entire port.

If you do not enable the recovery for the cause, the interface stays in the error-disabled state until you enter the **shutdown** and the **no shutdown** interface configuration commands. If you enable the recovery for a cause, the interface is brought out of the error-disabled state and allowed to retry the operation again when all the causes have timed out.

Otherwise, you must enter the **shutdown** and then the **no shutdown** commands to manually recover an interface from the error-disabled state.

You can verify your settings by entering the **show errdisable recovery** privileged EXEC command.

### Examples

This example shows how to enable the recovery timer for the BPDU guard error-disabled cause:

```
Device(config)# errdisable recovery cause bpduguard
```

## errdisable recovery cause small-frame

Use the **errdisable recovery cause small-frame** global configuration command on the switch to enable the recovery timer for ports to be automatically re-enabled after they are error disabled by the arrival of small frames. Use the **no** form of this command to return to the default setting.

**errdisable recovery cause small-frame**

**no errdisable recovery cause small-frame**

**Syntax Description** This command has no arguments or keywords.

**Command Default** This feature is disabled.

**Command Modes** Global configuration

Command History	Release	Modification
	Cisco IOS Release 15.0(2)EX	This command was introduced.

**Usage Guidelines** This command enables the recovery timer for error-disabled ports. You configure the recovery time by using the **errdisable recovery interval** interface configuration command.

This example shows how to set the recovery timer:

```
Device(config)# errdisable recovery cause small-frame
```



# errdisable recovery interval

To specify the time to recover from an error-disabled state, use the **errdisable recovery interval** command in global configuration mode. To return to the default setting, use the **no** form of this command.

**errdisable recovery interval** *timer-interval*  
**no errdisable recovery interval** *timer-interval*

<b>Syntax Description</b>	<i>timer-interval</i> Time to recover from the error-disabled state. The range is 30 to 86400 seconds. The same interval is applied to all causes. The default interval is 300 seconds.				
<b>Command Default</b>	The default recovery interval is 300 seconds.				
<b>Command Modes</b>	Global configuration				
<b>Command History</b>	<table border="1"> <thead> <tr> <th>Release</th> <th>Modification</th> </tr> </thead> <tbody> <tr> <td>Cisco IOS Release 15.0(2)EX</td> <td>This command was introduced.</td> </tr> </tbody> </table>	Release	Modification	Cisco IOS Release 15.0(2)EX	This command was introduced.
Release	Modification				
Cisco IOS Release 15.0(2)EX	This command was introduced.				

**Usage Guidelines** The error-disabled recovery timer is initialized at a random differential from the configured interval value. The difference between the actual timeout value and the configured value can be up to 15 percent of the configured interval.

You can verify your settings by entering the **show errdisable recovery** privileged EXEC command.

## Examples

This example shows how to set the timer to 500 seconds:

```
Device(config)# errdisable recovery interval 500
```

## lldp (interface configuration)

To enable Link Layer Discovery Protocol (LLDP) on an interface, use the **lldp** command in interface configuration mode. To disable LLDP on an interface, use the **no** form of this command.

Syntax Description		
<b>med-tlv-select</b>		Selects an LLDP Media Endpoint Discovery (MED) time-length-value (TLV) element to send.
<i>tlv</i>		String that identifies the TLV element. Valid values are the following: <ul style="list-style-type: none"> <li>• <b>inventory-management</b>— LLDP MED Inventory Management TLV.</li> <li>• <b>location</b>— LLDP MED Location TLV.</li> <li>• <b>network-policy</b>— LLDP MED Network Policy TLV.</li> </ul>
<b>receive</b>		Enables the interface to receive LLDP transmissions.
<b>tlv-select</b>		Selects the LLDP TLVs to send.
<b>power-management</b>		Sends the LLDP Power Management TLV.
<b>transmit</b>		Enables LLDP transmission on the interface.

**Command Default** LLDP is disabled.

**Command Modes** Interface configuration

Command History	Release	Modification
	Cisco IOS Release 15.0(2)EX	This command was introduced.

**Usage Guidelines** This command is supported on 802.1 media types.  
 If the interface is configured as a tunnel port, LLDP is automatically disabled.  
 The following example shows how to disable LLDP transmission on an interface:

```
Device(config)# interface gigabitethernet1/0/1
Device(config-if)# no lldp transmit
```

The following example shows how to enable LLDP transmission on an interface:

```
Device(config)# interface gigabitethernet1/0/1
Device(config-if)# lldp transmit
```

# mdix auto

To enable the automatic medium-dependent interface crossover (auto-MDIX) feature on the interface, use the **mdix auto** command in interface configuration mode. To disable auto-MDIX, use the **no** form of this command.

**mdix auto**  
**no mdix auto**

**Syntax Description** This command has no arguments or keywords.

**Command Default** Auto-MDIX is enabled.

**Command Modes** Interface configuration

Command History	Release	Modification
	Cisco IOS Release 15.0(2)EX	This command was introduced.

**Usage Guidelines** When auto-MDIX is enabled, the interface automatically detects the required cable connection type (straight-through or crossover) and configures the connection appropriately.

When you enable auto-MDIX on an interface, you must also set the interface speed and duplex to **auto** so that the feature operates correctly.

When auto-MDIX (and autonegotiation of speed and duplex) is enabled on one or both of the connected interfaces, link up occurs, even if the cable type (straight-through or crossover) is incorrect.

Auto-MDIX is supported on all 10/100 and 10/100/1000 Mb/s interfaces and on 10/100/1000BASE-TX small form-factor pluggable (SFP) module interfaces. It is not supported on 1000BASE-SX or -LX SFP module interfaces.

You can verify the operational state of auto-MDIX on the interface by entering the **show controllers ethernet-controller interface-id phy** privileged EXEC command.

This example shows how to enable auto-MDIX on a port:

```
Device# configure terminal
Device(config)# interface gigabitethernet1/0/1
Device(config-if)# speed auto
Device(config-if)# duplex auto
Device(config-if)# mdix auto
Device(config-if)# end
```

# network-policy

To apply a network-policy profile to an interface, use the **network-policy** command in interface configuration mode. To remove the policy, use the **no** form of this command.

**network-policy** *profile-number*  
**no network-policy**

## Syntax Description

*profile-number* The network-policy profile number to apply to the interface.

## Command Default

No network-policy profiles are applied.

## Command Modes

Interface configuration

## Command History

Release	Modification
Cisco IOS Release 15.0(2)EX	This command was introduced.

## Usage Guidelines

Use the **network-policy** *profile number* interface configuration command to apply a profile to an interface.

You cannot apply the **switchport voice vlan** command on an interface if you first configure a network-policy profile on it. However, if **switchport voice vlan** *vlan-id* is already configured on the interface, you can apply a network-policy profile on the interface. The interface then has the voice or voice-signaling VLAN network-policy profile applied.

This example shows how to apply network-policy profile 60 to an interface:

```
Device(config)# interface gigabitethernet1/0/1
Device(config-if)# network-policy 60
```

## network-policy profile (global configuration)

To create a network-policy profile and to enter network-policy configuration mode, use the **network-policy profile** command in global configuration mode. To delete the policy and to return to global configuration mode, use the **no** form of this command.

**network-policy profile** *profile-number*  
**no network-policy profile** *profile-number*

<b>Syntax Description</b>	<i>profile-number</i> Network-policy profile number. The range is 1 to 4294967295.	
<b>Command Default</b>	No network-policy profiles are defined.	
<b>Command Modes</b>	Global configuration	
<b>Command History</b>	<b>Release</b>	<b>Modification</b>
	Cisco IOS Release 15.0(2)EX	This command was introduced.
<b>Usage Guidelines</b>	<p>Use the <b>network-policy profile</b> global configuration command to create a profile and to enter network-policy profile configuration mode.</p> <p>To return to privileged EXEC mode from the network-policy profile configuration mode, enter the <b>exit</b> command.</p> <p>When you are in network-policy profile configuration mode, you can create the profile for voice and voice signaling by specifying the values for VLAN, class of service (CoS), differentiated services code point (DSCP), and tagging mode.</p> <p>These profile attributes are contained in the Link Layer Discovery Protocol for Media Endpoint Devices (LLDP-MED) network-policy time-length-value (TLV).</p> <p>This example shows how to create network-policy profile 60:</p> <pre>Device(config)# <b>network-policy profile 60</b> Device(config-network-policy)#</pre>	

## nmsp attachment suppress

To suppress the reporting of attachment information from a specified interface, use the **nmsp attachment suppress** command in interface configuration mode. To return to the default setting, use the **no** form of this command.

**nmsp attachment suppress**  
**no nmsp attachment suppress**

**Syntax Description** This command has no arguments or keywords.

**Command Default** None

**Command Modes** Interface configuration (config-if)

Command History	Release	Modification
	Cisco IOS Release 15.0(2)EX	This command was introduced.

**Usage Guidelines** Use the **nmsp attachment suppress** interface configuration command to configure an interface to not send location and attachment notifications to a Cisco Mobility Services Engine (MSE).



**Note** Attachment information is not supported in Cisco IOS XE Denali 16.1.1 and later releases.

This example shows how to configure an interface to not send attachment information to the MSE:

```
Device(config)# interface gigabitethernet1/0/1
Device(config-if)# nmsp attachment suppress
```

## power efficient-ethernet auto

To enable Energy Efficient Ethernet (EEE) for an interface, use the **power efficient-ethernet auto** command in interface configuration mode. To disable EEE on an interface, use the **no** form of this command.

**power efficient-ethernet auto**  
**no power efficient-ethernet auto**

**Syntax Description** This command has no arguments or keywords.

**Command Default** EEE is enabled

**Command Modes** Interface configuration

Command History	Release	Modification
	Cisco IOS Release 15.0(2)EX	This command was introduced.

**Usage Guidelines** You can enable EEE on devices that support low power idle (LPI) mode. Such devices can save power by entering LPI mode during periods of low utilization. In LPI mode, systems on both ends of the link can save power by shutting down certain services. EEE provides the protocol needed to transition into and out of LPI mode in a way that is transparent to upper layer protocols and applications.

The **power efficient-ethernet auto** command is available only if the interface is EEE capable. To check if an interface is EEE capable, use the **show eee capabilities EXEC** command.

When EEE is enabled, the device advertises and autonegotiates EEE to its link partner. To view the current EEE status for an interface, use the **show eee status EXEC** command.

This command does not require a license.

This example shows how to enable EEE for an interface:

```
Device(config-if) # power efficient-ethernet auto
Device(config-if) #
```

This example shows how to disable EEE for an interface:

```
Device(config-if) # no power efficient-ethernet auto
Device(config-if) #
```

## power inline

To configure the power management mode on Power over Ethernet (PoE) ports, use the **power inline** command in interface configuration mode. To return to the default settings, use the **no** form of this command.

```
power inline {auto [max max-wattage] | never | port priority {high | low} | static [max
max-wattage]}
no power inline {auto | never | port priority {high | low} | static [max max-wattage]}
```

Syntax Description		
<b>auto</b>		Enables powered-device detection. If enough power is available, automatically allocates power to the PoE port after device detection. Allocation is first-come, first-serve.
<b>max</b> <i>max-wattage</i>		(Optional) Limits the power allowed on the port. The range is 4000 to 30000 mW. If no value is specified, the maximum is allowed.
<b>never</b>		Disables device detection, and disables power to the port.
<b>port</b>		Configures the power priority of the port. The default priority is low.
<b>priority</b> { <b>high</b>   <b>low</b> }		Sets the power priority of the port. In case of a power supply failure, ports configured as low priority are turned off first and ports configured as high priority are turned off last. The default priority is low.
<b>static</b>		Enables powered-device detection. Pre-allocates (reserves) power for a port before the switch discovers the powered device. This action guarantees that the device connected to the interface receives enough power.

**Command Default** The default is **auto** (enabled).  
 The maximum wattage is 30,000 mW.  
 The default port priority is low.

**Command Default** Interface configuration



Command History	Release	Modification
	Cisco IOS Release 15.0(2)EX	This command was introduced.

### Usage Guidelines

This command is supported only on PoE-capable ports. If you enter this command on a port that does not support PoE, this error message appears:

```
Device(config)# interface gigabitethernet1/0/1
Device(config-if)# power inline auto
                        ^
% Invalid input detected at '^' marker.
```

In a switch stack, this command is supported on all ports in the stack that support PoE.

Use the **max** *max-wattage* option to disallow higher-power powered devices. With this configuration, when the powered device sends Cisco Discovery Protocol (CDP) messages requesting more power than the maximum wattage, the switch removes power from the port. If the powered-device IEEE class maximum is greater than the maximum wattage, the switch does not power the device. The power is reclaimed into the global power budget.



**Note** The switch never powers any class 0 or class 3 device if the **power inline max** *max-wattage* command is configured for less than 30 W.

If the switch denies power to a powered device (the powered device requests more power through CDP messages or if the IEEE class maximum is greater than the maximum wattage), the PoE port is in a power-deny state. The switch generates a system message, and the Oper column in the **show power inline** privileged EXEC command output shows *power-deny*.

Use the **power inline static** *max* *max-wattage* command to give a port high priority. The switch allocates PoE to a port configured in static mode before allocating power to a port configured in auto mode. The switch reserves power for the static port when it is configured rather than upon device discovery. The switch reserves the power on a static port even when there is no connected device and whether or not the port is in a shutdown or in a no shutdown state. The switch allocates the configured maximum wattage to the port, and the amount is never adjusted through the IEEE class or by CDP messages from the powered device. Because power is pre-allocated, any powered device that uses less than or equal to the maximum wattage is guaranteed power when it is connected to a static port. However, if the powered device IEEE class is greater than the maximum wattage, the switch does not supply power to it. If the switch learns through CDP messages that the powered device needs more than the maximum wattage, the powered device is shut down.

If the switch cannot pre-allocate power when a port is in static mode (for example, because the entire power budget is already allocated to other auto or static ports), this message appears: Command rejected: power inline static: pwr not available. The port configuration remains unchanged.

When you configure a port by using the **power inline auto** or the **power inline static** interface configuration command, the port autonegotiates by using the configured speed and duplex settings. This is necessary to determine the power requirements of the connected device (whether or not it is a powered device). After the power requirements have been determined, the switch hardcodes the interface by using the configured speed and duplex settings without resetting the interface.

When you configure a port by using the **power inline never** command, the port reverts to the configured speed and duplex settings.

If a port has a Cisco powered device connected to it, you should not use the **power inline never** command to configure the port. A false link-up can occur, placing the port in an error-disabled state.

Use the **power inline port priority {high | low}** command to configure the power priority of a PoE port. Powered devices connected to ports with low port priority are shut down first in case of a power shortage.

You can verify your settings by entering the **show power inline EXEC** command.

## Examples

This example shows how to enable detection of a powered device and to automatically power a PoE port on a switch:

```
Device(config)# interface gigabitethernet1/0/2
Device(config-if)# power inline auto
```

This example shows how to configure a PoE port on a switch to allow a class 1 or a class 2 powered device:

```
Device(config)# interface gigabitethernet1/0/2
Device(config-if)# power inline auto max 7000
```

This example shows how to disable powered-device detection and to not power a PoE port on a switch:

```
Device(config)# interface gigabitethernet1/0/2
Device(config-if)# power inline never
```

This example shows how to set the priority of a port to high, so that it would be one of the last ports to be shut down in case of power supply failure:

```
Device(config)# interface gigabitethernet1/0/2
Device(config-if)# power inline port priority high
```

# power inline consumption

To override the amount of power specified by the IEEE classification for a powered device, use the **power inline consumption** command in global or interface configuration to specify the wattage used by each device. To return to the default power setting, use the **no** form of this command.

**power inline consumption** [**default**] *wattage*  
**no power inline consumption** [**default**]

<b>Syntax Description</b>	<p><b>default</b> The <b>default</b> keyword appears only in the global configuration. The command has the same effect with or without the keyword.</p> <p><i>wattage</i> Specifies the power that the switch budgets for the port. The range is 4000 to 15400 mW.</p>				
<b>Command Default</b>	The default power on each Power over Ethernet (PoE) port is 15400 mW.				
<b>Command Modes</b>	Global configuration Interface configuration				
<b>Command History</b>	<table border="1"> <thead> <tr> <th>Release</th> <th>Modification</th> </tr> </thead> <tbody> <tr> <td>Cisco IOS Release 15.0(2)EX</td> <td>This command was introduced.</td> </tr> </tbody> </table>	Release	Modification	Cisco IOS Release 15.0(2)EX	This command was introduced.
Release	Modification				
Cisco IOS Release 15.0(2)EX	This command was introduced.				

<b>Usage Guidelines</b>	<p>This command is supported only on the LAN Base image.</p> <p>When Cisco powered devices are connected to PoE ports, the switch uses Cisco Discovery Protocol (CDP) to determine the <i>CDP-specific</i> power consumption of the devices, which is the amount of power to allocate based on the CDP messages. The switch adjusts the power budget accordingly. This does not apply to IEEE third-party powered devices. For these devices, when the switch grants a power request, the switch adjusts the power budget according to the powered-device IEEE classification. If the powered device is a class 0 (class status unknown) or a class 3, the switch budgets 15400 mW for the device, regardless of the CDP-specific amount of power needed.</p> <p>If the powered device reports a higher class than its CDP-specific consumption or does not support power classification (defaults to class 0), the switch can power fewer devices because it uses the IEEE class information to track the global power budget.</p> <p>With PoE+, powered devices use IEEE 802.3at and LLDP power with media dependent interface (MDI) type, length, and value descriptions (TLVs), Power-via-MDA TLVs, for negotiating power up to 30 W. Cisco pre-standard devices and Cisco IEEE powered devices can use CDP or the IEEE 802.3at power-via-MDI power negotiation mechanism to request power levels up to 30 W.</p>
-------------------------	---



**Note** The initial allocation for Class 0, Class 3, and Class 4 powered devices is 15.4 W. When a device starts up and uses CDP or LLDP to send a request for more than 15.4 W, it can be allocated up to the maximum of 30 W.

By using the **power inline consumption** *wattage* configuration command, you can override the default power requirement of the IEEE classification. The difference between what is mandated by the IEEE classification and what is actually needed by the device is reclaimed into the global power budget for use by additional devices. You can then extend the switch power budget and use it more effectively.

Before entering the **power inline consumption** *wattage* configuration command, we recommend that you enable policing of the real-time power consumption by using the **power inline police [action log]** interface configuration command.



**Caution** You should carefully plan your switch power budget and make certain not to oversubscribe the power supply.

When you enter the **power inline consumption default** *wattage* or the **no power inline consumption default** global configuration command, or the **power inline consumption** *wattage* or the **no power inline consumption** interface configuration command, this caution message appears.

```
%CAUTION: Interface Gi1/0/1: Misconfiguring the 'power inline consumption/allocation'
command may cause damage to the switch and void your warranty. Take precaution not to
oversubscribe the power supply.
It is recommended to enable power policing if the switch supports it.
Refer to documentation.
```



**Note** When you manually configure the power budget, you must also consider the power loss over the cable between the switch and the powered device.

For more information about the IEEE power classifications, see the “Configuring Interface Characteristics” chapter in the software configuration guide for this release.

This command is supported only on PoE-capable ports. If you enter this command on a switch or port that does not support PoE, an error message appears.

In a switch stack, this command is supported on all switches or ports in the stack that support PoE.

You can verify your settings by entering the **show power inline consumption** privileged EXEC command.

## Examples

This example shows how to use the command in global configuration mode to configure the switch to budget 5000 mW to each PoE port:

```
Device(config)# power inline consumption default 5000
%CAUTION: Interface Gi1/0/1: Misconfiguring the 'power inline consumption/allocation'
command may cause damage to the switch and void your warranty. Take precaution not to
oversubscribe the power supply.
It is recommended to enable power policing if the switch supports it.
Refer to documentation.
```

This example shows how to use the command in interface configuration mode to configure the switch to budget 12000 mW to the powered device connected to a specific PoE port:

```
Device(config)# interface gigabitethernet1/0/2
Device(config-if)# power inline consumption 12000
%CAUTION: Interface Gi1/0/2: Misconfiguring the 'power inline consumption/allocation'
```

command may cause damage to the switch and void your warranty. Take precaution not to oversubscribe the power supply.

It is recommended to enable power policing if the switch supports it.  
Refer to documentation.

## power inline police

To enable policing of real-time power consumption on a powered device, use the **power inline police** command in interface configuration mode. To disable this feature, use the **no** form of this command

```
power inline police [action {errdisable | log}]
no power inline police
```

### Syntax Description

<b>action errdisable</b>	(Optional) Configures the device to turn off power to the port if the real-time power consumption exceeds the maximum power allocation on the port. This is the default action.
<b>action log</b>	(Optional) Configures the device to generate a syslog message while still providing power to a connected device if the real-time power consumption exceeds the maximum power allocation on the port.

### Command Default

Policing of the real-time power consumption of the powered device is disabled.

### Command Modes

Interface configuration

### Command History

Release	Modification
Cisco IOS Release 15.0(2)EX	This command was introduced.

### Usage Guidelines

This command is supported only on the LAN Base image.

This command is supported only on Power over Ethernet (PoE)-capable ports. If you enter this command on a device or port that does not support PoE, an error message appears.

In a switch stack, this command is supported on all switches or ports in the stack that support PoE and real-time power-consumption monitoring.

When policing of the real-time power consumption is enabled, the device takes action when a powered device consumes more power than the allocated maximum amount.

When PoE is enabled, the device senses the real-time power consumption of the powered device. This feature is called *power monitoring* or *power sensing*. The device also polices the power usage with the *power policing* feature.

When power policing is enabled, the device uses one of the these values as the cutoff power on the PoE port in this order:

1. The user-defined power level that limits the power allowed on the port when you enter the **power inline auto max** *max-wattage* or the **power inline static max** *max-wattage* interface configuration command
2. The device automatically sets the power usage of the device by using CDP power negotiation or by the IEEE classification and LLDP power negotiation.

If you do not manually configure the cutoff-power value, the device automatically determines it by using CDP power negotiation or the device IEEE classification and LLDP power negotiation. If CDP or LLDP are not enabled, the default value of 30 W is applied. However without CDP or LLDP, the device does not allow devices to consume more than 15.4 W of power because values from 15400 to 30000 mW are only allocated based on CDP or LLDP requests. If a powered device consumes more than 15.4 W without CDP or LLDP

negotiation, the device might be in violation of the maximum current  $I_{max}$  limitation and might experience an  $I_{cut}$  fault for drawing more current than the maximum. The port remains in the fault state for a time before attempting to power on again. If the port continuously draws more than 15.4 W, the cycle repeats.

When a powered device connected to a PoE+ port restarts and sends a CDP or LLDP packet with a power TLV, the device locks to the power-negotiation protocol of that first packet and does not respond to power requests from the other protocol. For example, if the device is locked to CDP, it does not provide power to devices that send LLDP requests. If CDP is disabled after the device has locked on it, the device does not respond to LLDP power requests and can no longer power on any accessories. In this case, you should restart the powered device.

If power policing is enabled, the device polices power usage by comparing the real-time power consumption to the maximum power allocated on the PoE port. If the device uses more than the maximum power allocation (or *cutoff power*) on the port, the device either turns power off to the port, or the device generates a syslog message and updates the LEDs (the port LEDs are blinking amber) while still providing power to the device.

- To configure the device to turn off power to the port and put the port in the error-disabled state, use the **power inline police** interface configuration command.
- To configure the device to generate a syslog message while still providing power to the device, use the **power inline police action log** command.

If you do not enter the **action log** keywords, the default action is to shut down the port, turn off power to it, and put the port in the PoE error-disabled state. To configure the PoE port to automatically recover from the error-disabled state, use the **errdisable detect cause inline-power** global configuration command to enable error-disabled detection for the PoE cause and the **errdisable recovery cause inline-power interval interval** global configuration command to enable the recovery timer for the PoE error-disabled cause.



---

**Caution** If policing is disabled, no action occurs when the powered device consumes more than the maximum power allocation on the port, which could adversely affect the device.

---

You can verify your settings by entering the **show power inline police** privileged EXEC command.

### Examples

This example shows how to enable policing of the power consumption and configuring the device to generate a syslog message on the PoE port on a device:

```
Device(config)# interface gigabitethernet1/0/2
Device(config-if)# power inline police action log
```

# show eee

To display Energy Efficient Ethernet (EEE) information for an interface, use the **show eee** command in EXEC mode.

**show eee**{**counters** | **capabilities interface** *interface-id* | **status interface** *interface-id*}

Syntax Description		
<b>counters</b>		Displays EEE counters.
<b>capabilities</b>		Displays EEE capabilities for the specified interface.
<b>status</b>		Displays EEE status information for the specified interface.
<b>interface</b> <i>interface-id</i>		Specifies the interface for which to display EEE capabilities or status information.

**Command Default** None

**Command Modes** User EXEC  
Privileged EXEC

Command History	Release	Modification
	Cisco IOS Release 15.0(2)EX	This command was introduced.

**Usage Guidelines** You can enable EEE on devices that support low power idle (LPI) mode. Such devices can save power by entering LPI mode during periods of low power utilization. In LPI mode, systems on both ends of the link can save power by shutting down certain services. EEE provides the protocol needed to transition into and out of LPI mode in a way that is transparent to upper layer protocols and applications.

To check if an interface is EEE capable, use the **show eee capabilities** command. You can enable EEE on an interface that is EEE capable by using the **power efficient-ethernet auto** interface configuration command.

To view the EEE status, LPI status, and wake error count information for an interface, use the **show eee status** command.

This is an example of output from the **show eee counters** command:

```
Device# show eee counters
ASIC #0
-----
      LP Active 1G           : 0
      LP Transitioning 1G    : 0
      LP Active Tx 100M      : 0
      LP Transitioning Tx 100M : 0
      LP Active Rx 100M      : 0
      LP Transitioning Rx 100M : 0

ASIC #1
-----
```



```

LP Active 1G           : 0
LP Transitioning 1G   : 0
LP Active Tx 100M     : 0
LP Transitioning Tx 100M : 0
LP Active Rx 100M     : 0
LP Transitioning Rx 100M : 0

```

This is an example of output from the **show eee capabilities** command on an interface where EEE is enabled:

```

Device# show eee capabilities interface gigabitethernet1/0/1
Gi1/0/1
EEE(efficient-ethernet): yes (100-Tx and 1000T auto)
Link Partner           : yes (100-Tx and 1000T auto)

```

This is an example of output from the **show eee capabilities** command on an interface where EEE is not enabled:

```

Device# show eee capabilities interface gigabitethernet2/0/1
Gi2/0/1
EEE(efficient-ethernet): not enabled
Link Partner           : not enabled

```

This is an example of output from the **show eee status** command on an interface where EEE is enabled and operational. The table that follows describes the fields in the display.

```

Device# show eee status interface gigabitethernet1/0/4
Gi1/0/4 is up
EEE(efficient-ethernet): Operational
Rx LPI Status           : Received
Tx LPI Status           : Received

```

This is an example of output from the **show eee status** command on an interface where EEE is operational and the ports are in low power save mode:

```

Device# show eee status interface gigabitethernet1/0/3
Gi1/0/3 is up
EEE(efficient-ethernet): Operational
Rx LPI Status           : Low Power
Tx LPI Status           : Low Power
Wake Error Count        : 0

```

This is an example of output from the **show eee status** command on an interface where EEE is not enabled because a remote link partner is incompatible with EEE:

```

Device# show eee status interface gigabitethernet1/0/3
Gi1/0/3 is down
EEE(efficient-ethernet): Disagreed
Rx LPI Status           : None
Tx LPI Status           : None
Wake Error Count        : 0

```

Table 1: show eee status Field Descriptions

Field	Description
EEE (efficient-ethernet)	<p>The EEE status for the interface. This field can have any of the following values:</p> <ul style="list-style-type: none"> <li>• N/A—The port is not capable of EEE.</li> <li>• Disabled—The port EEE is disabled.</li> <li>• Disagreed—The port EEE is not set because a remote link partner might be incompatible with EEE; either it is not EEE capable, or its EEE setting is incompatible.</li> <li>• Operational—The port EEE is enabled and operating.</li> </ul> <p>If the interface speed is configured as 10 Mbps, EEE is disabled internally. When the interface speed moves back to auto, 100 Mbps or 1000 Mbps, EEE becomes active again.</p>
Rx/Tx LPI Status	<p>The Low Power Idle (LPI) status for the link partner. These fields can have any of the following values:</p> <ul style="list-style-type: none"> <li>• N/A—The port is not capable of EEE.</li> <li>• Interrupted—The link partner is in the process of moving to low power mode.</li> <li>• Low Power—The link partner is in low power mode.</li> <li>• None— EEE is disabled or not capable at the link partner side.</li> <li>• Received—The link partner is in low power mode and there is traffic activity.</li> </ul> <p>If an interface is configured as half-duplex, the LPI status is None, which means the interface cannot be in low power mode until it is configured as full-duplex.</p>
Wake Error Count	<p>The number of PHY wake-up faults that have occurred. A wake-up fault can occur when EEE is enabled and the connection to the link partner is broken.</p> <p>This information is useful for PHY debugging.</p>

## show env

To display fan, temperature, redundant power system (RPS) availability, and power information, use the **show env** command in EXEC mode.

```
show env {all | fan | power [{all | switch [stack-member-number]]} | rps | stack [stack-member-number] | temperature [status]}
```

Syntax Description		
<b>all</b>	Displays the fan and temperature environmental status and the status of the internal power supplies and the RPS.	
<b>fan</b>	Displays the switch fan status.	
<b>power</b>	Displays the internal power status of the active switch.	
<b>all</b>	(Optional) Displays the status of all the internal power supplies in a standalone switch when the command is entered on the switch, or in all the member switches when the command is entered on the active switch.	
<b>switch</b>	(Optional) Displays the status of the internal power supplies for each switch in the stack or for the specified switch.  This keyword is available only on stacking-capable switches.	
<i>stack-member-number</i>	(Optional) Number of the member switch for which to display the status of the internal power supplies or the environmental status.  The range is 1 to 8.	
<b>rps</b>	Displays the RPS status.	
<b>stack</b>	Displays all environmental status for each switch in the stack or for the specified switch.  This keyword is available only on stacking-capable switches.	
<b>temperature</b>	Displays the switch temperature status.	
<b>status</b>	(Optional) Displays the switch internal temperature (not the external temperature) and the threshold values.	
<b>Command Default</b>	None	
<b>Command Modes</b>	User EXEC  Privileged EXEC	
<b>Command History</b>	<b>Release</b>	<b>Modification</b>
	Cisco IOS Release 15.0(2)EX	This command was introduced.

**Usage Guidelines**

Use the **show env EXEC** command to display the information for the switch being accessed—a standalone switch or the active switch. Use this command with the **stack** and **switch** keywords to display all information for the stack or for the specified member switch.

If you enter the **show env temperature status** command, the command output shows the switch temperature state and the threshold level.

You can also use the **show env temperature** command to display the switch temperature status. The command output shows the green and yellow states as *OK* and the red state as *FAULTY*. If you enter the **show env all** command, the command output is the same as the **show env temperature status** command output.

**Examples**

This is an example of output from the **show env power all** command on the active switch:

*Table 2: States in the show env temperature status Command Output*

State	Description
Green	The switch temperature is in the <i>normal</i> operating range.
Yellow	The temperature is in the <i>warning</i> range. You should check the external temperature around the switch.
Red	The temperature is in the <i>critical</i> range. The switch might not run properly if the temperature is in this range.

# show errdisable detect

To display error-disabled detection status, use the **show errdisable detect** command in EXEC mode.

## show errdisable detect

**Syntax Description** This command has no arguments or keywords.

**Command Default** None

**Command Modes** User EXEC  
Privileged EXEC

Command History	Release	Modification
	Cisco IOS Release 15.0(2)EX	This command was introduced.

**Usage Guidelines** A gbic-invalid error reason refers to an invalid small form-factor pluggable (SFP) module.

The error-disable reasons in the command output are listed in alphabetical order. The mode column shows how error-disable is configured for each feature.

You can configure error-disabled detection in these modes:

- port mode—The entire physical port is error-disabled if a violation occurs.
- vlan mode—The VLAN is error-disabled if a violation occurs.
- port/vlan mode—The entire physical port is error-disabled on some ports and is per-VLAN error-disabled on other ports.

# show errdisable recovery

To display the error-disabled recovery timer information, use the **show errdisable recovery** command in EXEC mode.

## show errdisable recovery

**Syntax Description** This command has no arguments or keywords.

**Command Default** None

**Command Modes** User EXEC  
Privileged EXEC

Command History	Release	Modification
	Cisco IOS Release 15.0(2)EX	This command was introduced.

**Usage Guidelines** A gbic-invalid error-disable reason refers to an invalid small form-factor pluggable (SFP) module interface.



**Note** Though visible in the output, the unicast-flood field is not valid.

This is an example of output from the **show errdisable recovery** command:

# show interfaces

To display the administrative and operational status of all interfaces or for a specified interface, use the **show interfaces** command in privileged EXEC mode.

```
show interfaces [{interface-id | vlan vlan-id}] [{accounting | capabilities [module number] | debounce | description | etherchannel | flowcontrol | pruning | stats | status [{err-disabled}] | trunk}]
```

Syntax Description	
<i>interface-id</i>	(Optional) ID of the interface. Valid interfaces include physical ports (including type, stack member for stacking-capable switches, module, and port number) and port channels. The port channel range is 1 to 48.
<b>vlan</b> <i>vlan-id</i>	(Optional) VLAN identification. The range is 1 to 4094.
<b>accounting</b>	(Optional) Displays accounting information on the interface, including active protocols and input and output packets and octets.  <b>Note</b> The display shows only packets processed in software; hardware-switched packets do not appear.
<b>capabilities</b>	(Optional) Displays the capabilities of all interfaces or the specified interface, including the features and options that you can configure on the interface. Though visible in the command line help, this option is not available for VLAN IDs.
<b>module</b> <i>number</i>	(Optional) Displays capabilities of all interfaces on the switch or specified stack member.  The range is 1 to 8.  This option is not available if you entered a specific interface ID.
<b>debounce</b>	(Optional) Displays port debounce timer information for an interface.
<b>description</b>	(Optional) Displays the administrative status and description set for an interface.
<b>etherchannel</b>	(Optional) Displays interface EtherChannel information.
<b>flowcontrol</b>	(Optional) Displays interface flow control information.
<b>mtu</b>	(Optional) Displays the MTU for each interface or for the specified interface.
<b>pruning</b>	(Optional) Displays trunk VTP pruning information for the interface.
<b>stats</b>	(Optional) Displays the input and output packets by switching the path for the interface.

<b>status</b>	(Optional) Displays the status of the interface. A status of unsupported in the Type field means that a non-Cisco small form-factor pluggable (SFP) module is inserted in the module slot.
<b>err-disabled</b>	(Optional) Displays interfaces in an error-disabled state.
<b>trunk</b>	(Optional) Displays interface trunk information. If you do not specify an interface, only information for active trunking ports appears.



**Note** Though visible in the command-line help strings, the **crb**, **fair-queue**, **irb**, **mac-accounting**, **precedence**, **random-detect**, and **rate-limit** keywords are not supported.

**Command Default** None

**Command Modes** Privileged EXEC

<b>Command History</b>	<b>Release</b>	<b>Modification</b>
	Cisco IOS Release 15.0(2)EX	This command was introduced.

**Usage Guidelines** The **show interfaces capabilities** command with different keywords has these results:

- Use the **show interface capabilities module *number*** command to display the capabilities of all interfaces on that switch in the stack. If there is no with that module number in the stack, there is no output.
- Use the **show interfaces *interface-id* capabilities** to display the capabilities of the specified interface.
- Use the **show interfaces capabilities** (with no module number or interface ID) to display the capabilities of all interfaces in the stack.

This is an example of output from the **show interfaces** command for an interface on stack member 3:

```
Device# show interfaces gigabitethernet3/0/2
GigabitEthernet3/0/2 is down, line protocol is down (notconnect)
  Hardware is Gigabit Ethernet, address is 2037.064d.4381 (bia 2037.064d.4381)
  MTU 1500 bytes, BW 1000000 Kbit/sec, DLY 10 usec,
    reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ARPA, loopback not set
  Keepalive set (10 sec)
  Auto-duplex, Auto-speed, media type is 10/100/1000BaseTX
  input flow-control is off, output flow-control is unsupported
  ARP type: ARPA, ARP Timeout 04:00:00
  Last input never, output never, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/2000/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue: 0/40 (size/max)
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
    0 packets input, 0 bytes, 0 no buffer
    Received 0 broadcasts (0 multicasts)
```



```

0 runts, 0 giants, 0 throttles
0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored
0 watchdog, 0 multicast, 0 pause input
0 input packets with dribble condition detected
0 packets output, 0 bytes, 0 underruns
0 output errors, 0 collisions, 1 interface resets
0 unknown protocol drops
0 babbles, 0 late collision, 0 deferred
0 lost carrier, 0 no carrier, 0 pause output
0 output buffer failures, 0 output buffers swapped out

```

This is an example of output from the **show interfaces accounting** command:

```

Device# show interfaces accounting
Vlan1
      Protocol  Pkts In  Chars In  Pkts Out  Chars Out
      IP        382021  29073978  41157     20408734
      ARP        981     58860    179       10740
FastEthernet0
      Protocol  Pkts In  Chars In  Pkts Out  Chars Out
      Other     4        276      0         0
      Spanning Tree  41      2132    0         0
      CDP       5        2270    10        4318
GigabitEthernet1/0/1
      Protocol  Pkts In  Chars In  Pkts Out  Chars Out
No traffic sent or received on this interface.
GigabitEthernet1/0/2
      Protocol  Pkts In  Chars In  Pkts Out  Chars Out
No traffic sent or received on this interface.
GigabitEthernet1/0/3
      Protocol  Pkts In  Chars In  Pkts Out  Chars Out
      Other     0         0      226505    14949330
      Spanning Tree  679120  40747200  0         0
      CDP       22623   10248219  22656    10670858
      DTP      45226   2713560   0         0
GigabitEthernet1/0/4
      Protocol  Pkts In  Chars In  Pkts Out  Chars Out
No traffic sent or received on this interface.
GigabitEthernet1/0/5
      Protocol  Pkts In  Chars In  Pkts Out  Chars Out
No traffic sent or received on this interface.
GigabitEthernet1/0/6
      Protocol  Pkts In  Chars In  Pkts Out  Chars Out
No traffic sent or received on this interface.

<output truncated>

```

```

Device# show interfaces gigabitethernet1/0/1 capabilities
GigabitEthernet1/0/1
  Model: WS-C2960X-48TS-L
  Type: 10/100/1000BaseTX
  Speed: 10,100,1000,auto
  Duplex: half,full,auto
  Trunk encap. type: 802.1Q
  Trunk mode: on,off,desirable,nonegotiate
  Channel: yes
  Broadcast suppression: percentage(0-100)
  Flowcontrol: rx-(off,on,desired),tx-(none)
  Fast Start: yes
  QoS scheduling: rx-(not configurable on per port basis),
  tx-(4q3t) (3t: Two configurable values and one fixed.)
  CoS rewrite: yes

```

```

ToS rewrite:      yes
UDLD:            yes
Inline power:    no
SPAN:           source/destination
PortSecure:     yes
Dot1x:         yes

```

This is an example of output from the **show interfaces interface description** command when the interface has been described as *Connects to Marketing* by using the **description** interface configuration command:

```

Device# show interfaces gigabitethernet1/0/2 description
Interface          Status      Protocol Description
Gi1/0/2            up          down      Connects to Marketing

```

This is an example of output from the **show interfaces interface-id pruning** command when pruning is enabled in the VTP domain:

```

Device# show interfaces gigabitethernet1/0/2 pruning
Port          Vlans pruned for lack of request by neighbor
Gi1/0/2       3,4

Port          Vlans traffic requested of neighbor
Gi1/0/2       1-3

```

This is an example of output from the **show interfaces stats** command for a specified VLAN interface:

```

Device# show interfaces vlan 1 stats
Switching path  Pkts In   Chars In   Pkts Out   Chars Out
  Processor    1165354   136205310   570800     91731594
  Route cache      0         0           0           0
  Total         1165354   136205310   570800     91731594

```

This is an example of output from the **show interfaces interface-id status** command:

```

Device# show interfaces gigabitethernet1/0/20 status
Port      Name          Status      Vlan      Duplex  Speed      Type
Gi1/0/20          notconnect  1          auto     auto    10/100/1000Ba
seTX

```

This is an example of output from the **show interfaces status err-disabled** command. It displays the status of interfaces in the error-disabled state:

```

Device# show interfaces status err-disabled
Port      Name          Status      Reason
Gi1/0/2          err-disabled  gbic-invalid
Gi2/0/3          err-disabled  dtp-flap

```

This is an example of output from the **show interfaces interface-id pruning** command:

```

Device# show interfaces gigabitethernet1/0/2 pruning
Port Vlans pruned for lack of request by neighbor

Device# show interfaces gigabitethernet1/0/1 trunk
Port      Mode          Encapsulation  Status      Native vlan
Gi1/0/1   on            802.1q         other       10

Port      Vlans allowed on trunk
Gi1/0/1   none

```

```
Port      Vlans allowed and active in management domain
Gi1/0/1   none

Port      Vlans in spanning tree forwarding state and not pruned
Gi1/0/1   none
```

# show interfaces counters

To display various counters for the switch or for a specific interface, use the **show interfaces counters** command in privileged EXEC mode.

**show interfaces** [*interface-id*] **counters** [{**errors** | **etherchannel** | **module** *stack-member-number* | **protocol status** | **trunk**}]

## Syntax Description

<i>interface-id</i>	(Optional) ID of the physical interface, including type, stack member (stacking-capable switches only) module, and port number.
<b>errors</b>	(Optional) Displays error counters.
<b>etherchannel</b>	(Optional) Displays EtherChannel counters, including octets, broadcast packets, multicast packets, and unicast packets received and sent.
<b>module</b> <i>stack-member-number</i>	(Optional) Displays counters for the specified stack member. The range is 1 to 8.  <b>Note</b> In this command, the <b>module</b> keyword refers to the stack member number. The module number that is part of the interface ID is always zero.
<b>protocol status</b>	(Optional) Displays the status of protocols enabled on interfaces.
<b>trunk</b>	(Optional) Displays trunk counters.



**Note** Though visible in the command-line help string, the **vlan** *vlan-id* keyword is not supported.

## Command Default

None

## Command Modes

Privileged EXEC

## Command History

Release	Modification
Cisco IOS Release 15.0(2)EX	This command was introduced.

## Usage Guidelines

If you do not enter any keywords, all counters for all interfaces are included.

This is an example of partial output from the **show interfaces counters** command. It displays all counters for the switch.

```
Device# show interfaces counters
Port          InOctets    InUcastPkts  InMcastPkts  InBcastPkts
Gi1/0/1             0             0             0             0
Gi1/0/2             0             0             0             0
Gi1/0/3          95285341      43115         1178430       1950
Gi1/0/4             0             0             0             0
```

<output truncated>

This is an example of partial output from the **show interfaces counters module** command for stack member 2. It displays all counters for the specified switch in the stack.

```
Device# show interfaces counters module 2
Port          InOctets    InUcastPkts  InMcastPkts  InBcastPkts
Gi1/0/1       520         2            0            0
Gi1/0/2       520         2            0            0
Gi1/0/3       520         2            0            0
Gi1/0/4       520         2            0            0
```

<output truncated>

This is an example of partial output from the **show interfaces counters protocol status** command for all interfaces:

```
Device# show interfaces counters protocol status
Protocols allocated:
Vlan1: Other, IP
Vlan20: Other, IP, ARP
Vlan30: Other, IP, ARP
Vlan40: Other, IP, ARP
Vlan50: Other, IP, ARP
Vlan60: Other, IP, ARP
Vlan70: Other, IP, ARP
Vlan80: Other, IP, ARP
Vlan90: Other, IP, ARP
Vlan900: Other, IP, ARP
Vlan3000: Other, IP
Vlan3500: Other, IP
GigabitEthernet1/0/1: Other, IP, ARP, CDP
GigabitEthernet1/0/2: Other, IP
GigabitEthernet1/0/3: Other, IP
GigabitEthernet1/0/4: Other, IP
GigabitEthernet1/0/5: Other, IP
GigabitEthernet1/0/6: Other, IP
GigabitEthernet1/0/7: Other, IP
GigabitEthernet1/0/8: Other, IP
GigabitEthernet1/0/9: Other, IP
GigabitEthernet1/0/10: Other, IP, CDP
```

<output truncated>

This is an example of output from the **show interfaces counters trunk** command. It displays trunk counters for all interfaces.

```
Device# show interfaces counters trunk
Port          TrunkFramesTx  TrunkFramesRx  WrongEncap
Gi1/0/1       0              0              0
Gi1/0/2       0              0              0
Gi1/0/3       80678         0              0
Gi1/0/4       82320         0              0
Gi1/0/5       0              0              0
```

<output truncated>

# show interfaces switchport

To display the administrative and operational status of a switching (nonrouting) port, including port blocking and port protection settings, use the **show interfaces switchport** command in privileged EXEC mode.

**show interfaces** [*interface-id*] **switchport** [{**backup** [**detail**] | **module** *number*}]

<b>Syntax Description</b>	<i>interface-id</i>	(Optional) ID of the interface. Valid interfaces include physical ports (including type, stack member for stacking-capable switches, module, and port number) and port channels. The port channel range is 1 to 48.
	<b>backup</b>	(Optional) Displays Flex Link backup interface configuration for the specified interface or all interfaces.
	<b>detail</b>	(Optional) Displays detailed backup information for the specified interface or all interfaces on the switch or the stack.
	<b>module</b> <i>number</i>	(Optional) Displays switchport configuration of all interfaces on the switch or specified stack member. The range is 1 to 8. This option is not available if you entered a specific interface ID.
<b>Command Default</b>	None	
<b>Command Modes</b>	Privileged EXEC	
<b>Command History</b>	<b>Release</b>	<b>Modification</b>
	Cisco IOS Release 15.0(2)EX	This command was introduced.
<b>Usage Guidelines</b>	Use the <b>show interface switchport module</b> <i>number</i> command to display the switch port characteristics of all interfaces on that switch in the stack. If there is no switch with that module number in the stack, there is no output.	

This is an example of output from the **show interfaces switchport** command for a port. The table that follows describes the fields in the display.



**Note** Private VLANs are not supported in this release, so those fields are not applicable.

```
Device# show interfaces gigabitethernet1/0/1 switchport
Name: Gi1/0/1
Switchport: Enabled
Administrative Mode: trunk
Operational Mode: down
Administrative Trunking Encapsulation: dot1q
Negotiation of Trunking: On
Access Mode VLAN: 1 (default)
Trunking Native Mode VLAN: 10 (VLAN0010)
```

```

Administrative Native VLAN tagging: enabled
Voice VLAN: none
Administrative private-vlan host-association: none
Administrative private-vlan mapping: none
Administrative private-vlan trunk native VLAN: none
Administrative private-vlan trunk Native VLAN tagging: enabled
Administrative private-vlan trunk encapsulation: dot1q
Administrative private-vlan trunk normal VLANs: none
Administrative private-vlan trunk associations: none
Administrative private-vlan trunk mappings: none
Operational private-vlan: none
Trunking VLANs Enabled: 11-20
Pruning VLANs Enabled: 2-1001
Capture Mode Disabled
Capture VLANs Allowed: ALL

Protected: false
Unknown unicast blocked: disabled
Unknown multicast blocked: disabled
Appliance trust: none

```

Field	Description
Name	Displays the port name.
Switchport	Displays the administrative and operational status of the port. In this display, the port is in switchport mode.
Administrative Mode Operational Mode	Displays the administrative and operational modes.
Administrative Trunking Encapsulation Operational Trunking Encapsulation Negotiation of Trunking	Displays the administrative and operational encapsulation method and whether trunking negotiation is enabled.
Access Mode VLAN	Displays the VLAN ID to which the port is configured.
Trunking Native Mode VLAN Trunking VLANs Enabled Trunking VLANs Active	Lists the VLAN ID of the trunk that is in native mode. Lists the allowed VLANs on the trunk. Lists the active VLANs on the trunk.
Pruning VLANs Enabled	Lists the VLANs that are pruning-eligible.
Protected	Displays whether or not protected port is enabled (True) or disabled (False) on the interface.
Unknown unicast blocked Unknown multicast blocked	Displays whether or not unknown multicast and unknown unicast traffic is blocked on the interface.
Voice VLAN	Displays the VLAN ID on which voice VLAN is enabled.

Field	Description
Appliance trust	Displays the class of service (CoS) setting of the data packets of the IP phone.

This is an example of output from the **show interfaces switchport backup** command:

```
Device# show interfaces switchport backup
Switch Backup Interface Pairs:
Active Interface   Backup Interface   State
-----
Gi1/0/1           Gi1/0/2           Active Up/Backup Standby
Gi3/0/3           Gi4/0/5           Active Down/Backup Up
Po1               Po2               Active Standby/Backup Up
```

In this example of output from the **show interfaces switchport backup** command, VLANs 1 to 50, 60, and 100 to 120 are configured on the switch:

```
Device(config)# interface gigabitethernet 2/0/6
Device(config-if)# switchport backup interface gigabitethernet 2/0/8
prefer vlan 60,100-120
```

When both interfaces are up, Gi2/0/8 forwards traffic for VLANs 60, 100 to 120, and Gi2/0/6 will forward traffic for VLANs 1 to 50.

```
Device# show interfaces switchport backup

Switch Backup Interface Pairs:
Active Interface   Backup Interface   State
-----
GigabitEthernet2/0/6  GigabitEthernet2/0/8  Active Up/Backup Up
Vlans on Interface Gi 2/0/6: 1-50
Vlans on Interface Gi 2/0/8: 60, 100-120
```

When a Flex Link interface goes down (LINK\_DOWN), VLANs preferred on this interface are moved to the peer interface of the Flex Link pair. In this example, if interface Gi2/0/6 goes down, Gi2/0/8 carries all VLANs of the Flex Link pair.

```
Device# show interfaces switchport backup

Switch Backup Interface Pairs:
Active Interface   Backup Interface   State
-----
GigabitEthernet2/0/6  GigabitEthernet2/0/8  Active Down/Backup Up
Vlans on Interface Gi 2/0/6:
Vlans on Interface Gi 2/0/8: 1-50, 60, 100-120
```

When a Flex Link interface comes up, VLANs preferred on this interface are blocked on the peer interface and moved to the forwarding state on the interface that has just come up. In this example, if interface Gi2/0/6 comes up, then VLANs preferred on this interface are blocked on the peer interface Gi2/0/8 and forwarded on Gi2/0/6.

```
Device# show interfaces switchport backup

Switch Backup Interface Pairs:
Active Interface   Backup Interface   State
-----
GigabitEthernet2/0/6  GigabitEthernet2/0/8  Active Up/Backup Up
Vlans on Interface Gi 2/0/6: 1-50
```



Vlans on Interface Gi 2/0/8: 60, 100-120

## show interfaces transceiver

To display the physical properties of a small form-factor pluggable (SFP) module interface, use the **show interfaces transceiver** command in EXEC mode.

**show interfaces** [*interface-id*] **transceiver** [{**detail** | **module number** | **properties** | **supported-list** | **threshold-table**}]

Syntax Description	
<i>interface-id</i>	(Optional) ID of the physical interface, including type, stack member (stacking-capable switches only) module, and port number.
<b>detail</b>	(Optional) Displays calibration properties, including high and low numbers and any alarm information for any Digital Optical Monitoring (DoM)-capable transceiver if one is installed in the switch.
<b>module number</b>	(Optional) Limits display to interfaces on module on the switch. The range is 1 to 8. This option is not available if you entered a specific interface ID.
<b>properties</b>	(Optional) Displays speed, duplex, and inline power settings on an interface.
<b>supported-list</b>	(Optional) Lists all supported transceivers.
<b>threshold-table</b>	(Optional) Displays alarm and warning threshold table.

Command Modes	
	User EXEC
	Privileged EXEC

Command History	Release	Modification
	Cisco IOS Release 15.0(2)EX	This command was introduced.

### Examples

This is an example of output from the **show interfaces *interface-id* transceiver properties** command:

```
Device# show interfaces gigabitethernet1/0/50 transceiver properties
Diagnostic Monitoring is not implemented.
Name : Gi1/0/50
Administrative Speed: auto
Administrative Duplex: auto
Administrative Auto-MDIX: on
Administrative Power Inline: N/A
Operational Speed: 1000
Operational Duplex: full
Operational Auto-MDIX: on
Media Type: 10/100/1000BaseTX
```

This is an example of output from the **show interfaces *interface-id* transceiver detail** command:

```

Device# show interfaces gigabitethernet1/1/1 transceiver detail
ITU Channel not available (Wavelength not available),
Transceiver is internally calibrated.
mA:milliamperes, dBm:decibels (milliwatts), N/A:not applicable.
++:high alarm, +:high warning, -:low warning, -- :low alarm.
A2D readouts (if they differ), are reported in parentheses.
The threshold values are uncalibrated.

```

Port	Temperature (Celsius)	High Alarm Threshold (Celsius)	High Warn Threshold (Celsius)	Low Warn Threshold (Celsius)	Low Alarm Threshold (Celsius)
Gil/1/1	29.9	74.0	70.0	0.0	-4.0
Port	Voltage (Volts)	High Alarm Threshold (Volts)	High Warn Threshold (Volts)	Low Warn Threshold (Volts)	Low Alarm Threshold (Volts)
Gil/1/1	3.28	3.60	3.50	3.10	3.00
Port	Optical Transmit Power (dBm)	High Alarm Threshold (dBm)	High Warn Threshold (dBm)	Low Warn Threshold (dBm)	Low Alarm Threshold (dBm)
Gil/1/1	1.8	7.9	3.9	0.0	-4.0
Port	Optical Receive Power (dBm)	High Alarm Threshold (dBm)	High Warn Threshold (dBm)	Low Warn Threshold (dBm)	Low Alarm Threshold (dBm)
Gil/1/1	-23.5	-5.0	-9.0	-28.2	-32.2

This is an example of output from the **show interfaces transceiver threshold-table** command:

```

Device# show interfaces transceiver threshold-table

```

	Optical Tx	Optical Rx	Temp	Laser Bias current	Voltage
DWDM GBIC					
Min1	-4.00	-32.00	-4	N/A	4.65
Min2	0.00	-28.00	0	N/A	4.75
Max2	4.00	-9.00	70	N/A	5.25
Max1	7.00	-5.00	74	N/A	5.40
DWDM SFP					
Min1	-4.00	-32.00	-4	N/A	3.00
Min2	0.00	-28.00	0	N/A	3.10
Max2	4.00	-9.00	70	N/A	3.50
Max1	8.00	-5.00	74	N/A	3.60
RX only WDM GBIC					
Min1	N/A	-32.00	-4	N/A	4.65
Min2	N/A	-28.30	0	N/A	4.75
Max2	N/A	-9.00	70	N/A	5.25
Max1	N/A	-5.00	74	N/A	5.40
DWDM XENPAK					
Min1	-5.00	-28.00	-4	N/A	N/A
Min2	-1.00	-24.00	0	N/A	N/A
Max2	3.00	-7.00	70	N/A	N/A
Max1	7.00	-3.00	74	N/A	N/A
DWDM X2					
Min1	-5.00	-28.00	-4	N/A	N/A
Min2	-1.00	-24.00	0	N/A	N/A
Max2	3.00	-7.00	70	N/A	N/A

## show interfaces transceiver

Max1	7.00	-3.00	74	N/A	N/A
DWDM XFP					
Min1	-5.00	-28.00	-4	N/A	N/A
Min2	-1.00	-24.00	0	N/A	N/A
Max2	3.00	-7.00	70	N/A	N/A
Max1	7.00	-3.00	74	N/A	N/A
CWDM X2					
Min1	N/A	N/A	0	N/A	N/A
Min2	N/A	N/A	0	N/A	N/A
Max2	N/A	N/A	0	N/A	N/A
Max1	N/A	N/A	0	N/A	N/A

<output truncated>

# show ip ports all

To display all the open ports on the device, use the **show ip ports all** command in EXEC or User EXEC mode.

## show ip ports all

**Syntax Description** This command has no arguments or keywords.

**Command Default** None

**Command Modes** User EXEC, Privileged EXEC

Command History	Release	Modification
	15.2(5) E1	This command was introduced.

The following is a sample output from **show ip ports all** command:

```
switch# show ip ports all
TCB Proto Local Address Foreign Address State PID/Program Name
Local Address Foreign Address (state)
tcp *:4786 *:*: LISTEN 224/[IOS]SMI IBC server process
tcp *:443 *:*: LISTEN 286/[IOS]HTTP CORE
tcp *:443 *:*: LISTEN 286/[IOS]HTTP CORE
tcp *:80 *:*: LISTEN 286/[IOS]HTTP CORE
tcp *:80 *:*: LISTEN 286/[IOS]HTTP CORE
udp *:10002 *:*: 0/[IOS] Unknown
udp *:2228 0.0.0.0:0 318/[IOS]L2TRACE SERVER
```

switch#

The table below shows the field descriptions.

Field	Description
Protocol	Transport protocol used
Foreign Address	Remote / peer address
State	State of connection : listen / establishment / connected
PID/Program Name	Process id / process name
Local Address	Device IP address

**Related Commands** **show tcp brief all**

**show ip sockets**

# show network-policy profile

To display the network-policy profiles, use the **show network policy profile** command in privileged EXEC mode.

**show network-policy profile** [*profile-number*]

<b>Syntax Description</b>	<i>profile-number</i> (Optional) Displays the network-policy profile number. If no profile is entered, all network-policy profiles appear.	
<b>Command Default</b>	None	
<b>Command Modes</b>	Privileged EXEC	
<b>Command History</b>	<b>Release</b>	<b>Modification</b>
	Cisco IOS Release 15.0(2)EX	This command was introduced.

This is an example of output from the **show network-policy profile** command:

```
Device# show network-policy profile
Network Policy Profile 60
  Interface:
    none
```

# show power inline

To display the Power over Ethernet (PoE) status for the specified PoE port, the specified stack member, or for all PoE ports in the switch stack, use the **show power inline** command in EXEC mode.

**show power inline** [{**police** | **priority**}] [{*interface-id* | **module** *stack-member-number*}] [**detail**]

Syntax Description	
<b>police</b>	(Optional) Displays the power policing information about real-time power consumption.
<b>priority</b>	(Optional) Displays the power inline port priority for each port.
<i>interface-id</i>	(Optional) ID of the physical interface.
<b>module</b> <i>stack-member-number</i>	(Optional) Limits the display to ports on the specified stack member.  The range is 1 to 8.  This keyword is supported only on stacking-capable switches.
<b>detail</b>	(Optional) Displays detailed output of the interface or module.

Command Modes	
	User EXEC
	Privileged EXEC

Command History	Release	Modification
	Cisco IOS Release 15.0(2)EX	This command was introduced.

## Examples

This is an example of output from the **show power inline** command. The table that follows describes the output fields.

```
Device> show power inline
Module   Available   Used   Remaining
         (Watts)    (Watts) (Watts)
-----
1         n/a         n/a    n/a
2         n/a         n/a    n/a
3         1440.0     15.4   1424.6
4         720.0      6.3    713.7
Interface Admin Oper   Power Device      Class Max
         (Watts)
-----
Gi3/0/1  auto  off   0.0   n/a         n/a  30.0
Gi3/0/2  auto  off   0.0   n/a         n/a  30.0
Gi3/0/3  auto  off   0.0   n/a         n/a  30.0
Gi3/0/4  auto  off   0.0   n/a         n/a  30.0
Gi3/0/5  auto  off   0.0   n/a         n/a  30.0
Gi3/0/6  auto  off   0.0   n/a         n/a  30.0
Gi3/0/7  auto  off   0.0   n/a         n/a  30.0
Gi3/0/8  auto  off   0.0   n/a         n/a  30.0
```

## show power inline

```

Gi3/0/9   auto   off      0.0    n/a          n/a    30.0
Gi3/0/10  auto   off      0.0    n/a          n/a    30.0
Gi3/0/11  auto   off      0.0    n/a          n/a    30.0
Gi3/0/12  auto   off      0.0    n/a          n/a    30.0
<output truncated>

```

This is an example of output from the **show power inline interface-id** command on a switch port:

This is an example of output from the **show power inline module switch-number** command on stack member 3. The table that follows describes the output fields.

```

Device> show power inline module 3
Module   Available      Used      Remaining
         (Watts)        (Watts)   (Watts)
-----
3        865.0          864.0     1.0
Interface Admin  Oper      Power    Device   Class Max
         (Watts)
-----
Gi3/0/1  auto  power-deny 4.0     n/a      n/a    15.4
Gi3/0/2  auto  off        0.0     n/a      n/a    15.4
Gi3/0/3  auto  off        0.0     n/a      n/a    15.4
Gi3/0/4  auto  off        0.0     n/a      n/a    15.4
Gi3/0/5  auto  off        0.0     n/a      n/a    15.4
Gi3/0/6  auto  off        0.0     n/a      n/a    15.4
Gi3/0/7  auto  off        0.0     n/a      n/a    15.4
Gi3/0/8  auto  off        0.0     n/a      n/a    15.4
Gi3/0/9  auto  off        0.0     n/a      n/a    15.4
Gi3/0/10 auto  off        0.0     n/a      n/a    15.4
<output truncated>

```

**Table 3: show power inline Field Descriptions**

Field	Description
Available	The total amount of configured power <sup>1</sup> on the PoE switch in watts (W).
Used	The amount of configured power that is allocated to PoE ports in watts.
Remaining	The amount of configured power in watts that is not allocated to ports in the system. (Available – Used = Remaining)
Admin	Administration mode: auto, off, static.
Oper	Operating mode: <ul style="list-style-type: none"> <li>• on—The powered device is detected, and power is applied.</li> <li>• off—No PoE is applied.</li> <li>• faulty—Device detection or a powered device is in a faulty state.</li> <li>• power-deny—A powered device is detected, but no PoE is available, or the maximum wattage exceeds the detected powered-device maximum.</li> </ul>
Power	The maximum amount of power that is allocated to the powered device in watts. This value is the same as the value in the <i>Cutoff Power</i> field in the <b>show power inline police</b> command output.



Field	Description
Device	The device type detected: n/a, unknown, Cisco powered-device, IEEE powered-device, or the name from CDP.
Class	The IEEE classification: n/a or a value from 0 to 4.
Max	The maximum amount of power allocated to the powered device in watts.
AdminPowerMax	The maximum amount power allocated to the powered device in watts when the switch polices the real-time power consumption. This value is the same as the <i>Max</i> field value.
AdminConsumption	The power consumption of the powered device in watts when the switch polices the real-time power consumption. If policing is disabled, this value is the same as the <i>AdminPowerMax</i> field value.

- <sup>1</sup> The configured power is the power that you manually specify or that the switch specifies by using CDP power negotiation or the IEEE classification, which is different than the real-time power that is monitored with the power sensing feature.

This is an example of output from the **show power inline police** command on a stacking-capable switch:

```
Device> show power inline police
Module   Available   Used         Remaining
         (Watts)    (Watts)     (Watts)
-----
1         370.0      0.0         370.0
3         865.0      864.0       1.0

Interface Admin Oper   Admin   Oper   Cutoff Oper
         State State  Police  Police Power  Power
-----
Gi1/0/1  auto  off   none    n/a    n/a    0.0
Gi1/0/2  auto  off   log     n/a    5.4    0.0
Gi1/0/3  auto  off   errdisable n/a    5.4    0.0
Gi1/0/4  off   off   none    n/a    n/a    0.0
Gi1/0/5  off   off   log     n/a    5.4    0.0
Gi1/0/6  off   off   errdisable n/a    5.4    0.0
Gi1/0/7  auto  off   none    n/a    n/a    0.0
Gi1/0/8  auto  off   log     n/a    5.4    0.0
Gi1/0/9  auto  on    none    n/a    n/a    5.1
Gi1/0/10 auto  on    log     ok     5.4    4.2
Gi1/0/11 auto  on    log     log    5.4    5.9
Gi1/0/12 auto  on    errdisable ok     5.4    4.2
Gi1/0/13 auto  errdisable errdisable n/a    5.4    0.0
<output truncated>
```

In the previous example:

- The Gi1/0/1 port is shut down, and policing is not configured.
- The Gi1/0/2 port is shut down, but policing is enabled with a policing action to generate a syslog message.
- The Gi1/0/3 port is shut down, but policing is enabled with a policing action is to shut down the port.
- Device detection is disabled on the Gi1/0/4 port, power is not applied to the port, and policing is disabled.

- Device detection is disabled on the Gi1/0/5 port, and power is not applied to the port, but policing is enabled with a policing action to generate a syslog message.
- Device detection is disabled on the Gi1/0/6 port, and power is not applied to the port, but policing is enabled with a policing action to shut down the port.
- The Gi1/0/7 port is up, and policing is disabled, but the switch does not apply power to the connected device.
- The Gi1/0/8 port is up, and policing is enabled with a policing action to generate a syslog message, but the switch does not apply power to the powered device.
- The Gi1/0/9 port is up and connected to a powered device, and policing is disabled.
- The Gi1/0/10 port is up and connected to a powered device, and policing is enabled with a policing action to generate a syslog message. The policing action does not take effect because the real-time power consumption is less than the cutoff value.
- The Gi1/0/11 port is up and connected to a powered device, and policing is enabled with a policing action to generate a syslog message.
- The Gi1/0/12 port is up and connected to a powered device, and policing is enabled with a policing action to shut down the port. The policing action does not take effect because the real-time power consumption is less than the cutoff value.
- The Gi1/0/13 port is up and connected to a powered device, and policing is enabled with a policing action to shut down the port.

This is an example of output from the **show power inline police interface-id** command on a standalone switch. The table that follows describes the output fields.

**Table 4: show power inline police Field Descriptions**

Field	Description
Available	The total amount of configured power <sup>2</sup> on the switch in watts (W).
Used	The amount of configured power allocated to PoE ports in watts.
Remaining	The amount of configured power in watts that is not allocated to ports in the system. (Available – Used = Remaining)
Admin State	Administration mode: auto, off, static.
Oper State	<p>Operating mode:</p> <ul style="list-style-type: none"> <li>• errdisable—Policing is enabled.</li> <li>• faulty—Device detection on a powered device is in a faulty state.</li> <li>• off—No PoE is applied.</li> <li>• on—The powered device is detected, and power is applied.</li> <li>• power-deny—A powered device is detected, but no PoE is available, or the real-time power consumption exceeds the maximum power allocation.</li> </ul> <p><b>Note</b> The operating mode is the current PoE state for the specified PoE port, the specified stack member, or for all PoE ports on the switch.</p>

Field	Description
Admin Police	Status of the real-time power-consumption policing feature: <ul style="list-style-type: none"> <li>errdisable—Policing is enabled, and the switch shuts down the port when the real-time power consumption exceeds the maximum power allocation.</li> <li>log—Policing is enabled, and the switch generates a syslog message when the real-time power consumption exceeds the maximum power allocation.</li> <li>none—Policing is disabled.</li> </ul>
Oper Police	Policing status: <ul style="list-style-type: none"> <li>errdisable—The real-time power consumption exceeds the maximum power allocation, and the switch shuts down the PoE port.</li> <li>log—The real-time power consumption exceeds the maximum power allocation, and the switch generates a syslog message.</li> <li>n/a—Device detection is disabled, power is not applied to the PoE port, or no policing action is configured.</li> <li>ok—Real-time power consumption is less than the maximum power allocation.</li> </ul>
Cutoff Power	The maximum power allocated on the port. When the real-time power consumption is greater than this value, the switch takes the configured policing action.
Oper Power	The real-time power consumption of the powered device.

<sup>2</sup> The configured power is the power that you manually specify or that the switch specifies by using CDP power negotiation or the IEEE classification, which is different than the real-time power that is monitored with the power sensing feature.

# show system mtu

To display the global maximum transmission unit (MTU) or maximum packet size set for the switch, use the **show system mtu** command in privileged EXEC mode.

```
show system mtu
```

---

**Syntax Description** This command has no arguments or keywords.

---

**Command Default** None

---

**Command Modes** Privileged EXEC

---

Command History	Release	Modification
	Cisco IOS Release 15.0(2)EX	This command was introduced.

---



---

**Usage Guidelines** For information about the MTU values and the stack configurations that affect the MTU values, see the **system mtu** command.

---

**Examples** This is an example of output from the **show system mtu** command:

```
Device# show system mtu

System MTU size is 1500 bytes
System Jumbo MTU size is 1500 bytes
System Alternate MTU size is 1500 bytes
Routing MTU size is 1500 bytes
```

# speed

To specify the speed of a 10/100/1000/2500/5000 Mbps port, use the **speed** command in interface configuration mode. To return to the default value, use the **no** form of this command.

```
speed {10 | 100 | 1000 | 2500 | 5000 | auto [{10 | 100 | 1000 | 2500 | 5000}] | nonegotiate}
no speed
```

Syntax Description		
<b>10</b>	Specifies that the port runs at 10 Mbps.	
<b>100</b>	Specifies that the port runs at 100 Mbps.	
<b>1000</b>	Specifies that the port runs at 1000 Mbps. This option is valid and visible only on 10/100/1000 Mb/s ports.	
<b>2500</b>	Specifies that the port runs at 2500 Mbps. This option is valid and visible only on multi-Gigabit-supported Ethernet ports.	
<b>5000</b>	Specifies that the port runs at 5000 Mbps. This option is valid and visible only on multi-Gigabit-supported Ethernet ports.	
<b>auto</b>	Detects the speed at which the port should run, automatically, based on the port at the other end of the link. If you use the <b>10</b> , <b>100</b> , <b>1000</b> , <b>1000</b> , <b>2500</b> , or <b>5000</b> keyword with the <b>auto</b> keyword, the port autonegotiates only at the specified speeds.	
<b>nonegotiate</b>	Disables autonegotiation, and the port runs at 1000 Mbps.	

**Command Default** The default is **auto**.

**Command Modes** Interface configuration

Command History	Release	Modification
	Cisco IOS Release 15.0(2)EX	This command was introduced.

**Usage Guidelines** You cannot configure speed on 10-Gigabit Ethernet ports.

Except for the 1000BASE-T small form-factor pluggable (SFP) modules, you can configure the speed to not negotiate (**nonegotiate**) when an SFP module port is connected to a device that does not support autonegotiation.

The new keywords, **2500** and **5000** are visible only on multi-Gigabit (m-Gig) Ethernet supporting devices.

If the speed is set to **auto**, the switch negotiates with the device at the other end of the link for the speed setting, and then forces the speed setting to the negotiated value. The duplex setting remains configured on each end of the link, which might result in a duplex setting mismatch.

If both ends of the line support autonegotiation, we highly recommend the default autonegotiation settings. If one interface supports autonegotiation and the other end does not, use the auto setting on the supported side, but set the duplex and speed on the other side.



---

**Caution** Changing the interface speed and duplex mode configuration might shut down and re-enable the interface during the reconfiguration.

---

For guidelines on setting the switch speed and duplex parameters, see the “Configuring Interface Characteristics” chapter in the software configuration guide for this release.

Verify your settings using the **show interfaces** privileged EXEC command.

---

## Examples

The following example shows how to set speed on a port to 100 Mbps:

```
Device(config)# interface gigabitethernet1/0/1
Device(config-if)# speed 100
```

The following example shows how to set a port to autonegotiate at only 10 Mbps:

```
Device(config)# interface gigabitethernet1/0/1
Device(config-if)# speed auto 10
```

The following example shows how to set a port to autonegotiate at only 10 or 100 Mbps:

```
Device(config)# interface gigabitethernet1/0/1
Device(config-if)# speed auto 10 100
```

# switchport backup interface

To configure Flex Links, use the **switchport backup interface** command in interface configuration mode on a Layer 2 interface on the switch stack or on a standalone switch. To remove the Flex Links configuration, use the **no** form of this command.

```
switchport backup interface interface-id [{mmu primary vlan vlan-id|multicast fast-convergence
|preemption {delay seconds|mode {bandwidth|forced|off}}|prefer vlan vlan-id}]
no switchport backup interface interface-id [{mmu primary vlan|multicast fast-convergence|
preemption {delay|mode}|prefer vlan}]
```

Syntax Description		
	<i>interface-id</i>	ID of the physical interface.
	<b>mmu</b>	(Optional) Configures the MAC move update (MMU) for a backup interface pair.
	<b>primary vlan</b> <i>vlan-id</i>	(Optional) VLAN ID of the primary VLAN. The range is 1 to 4094.
	<b>multicast fast-convergence</b>	(Optional) Configures multicast fast convergence on the backup interface.
	<b>preemption</b>	(Optional) Configures a preemption scheme for a backup interface pair.
	<b>delay</b> <i>seconds</i>	Specifies a preemption delay. The range is 1 to 300 seconds. The default is 35 seconds.
	<b>mode</b>	Specifies the preemption mode.
	<b>bandwidth</b>	Specifies that a higher bandwidth interface is preferred.
	<b>forced</b>	Specifies that an active interface is preferred.
	<b>off</b>	Specifies that no preemption occurs from backup to active.
	<b>prefer vlan</b> <i>vlan-id</i>	(Optional) Specifies that VLANs are carried on the backup interfaces of a Flex Link pair. VLAN ID range is 1 to 4094.

**Command Default** The default is to have no Flex Links defined. The preemption mode is off. No preemption occurs. Preemption delay is set to 35 seconds.

**Command Modes** Interface configuration

Command History	Release	Modification
	Cisco IOS Release 15.0(2)EX	This command was introduced.

**Usage Guidelines** Flex Links are a pair of interfaces that provide backup to each other. With Flex Links configured, one link acts as the primary interface and forwards traffic, while the other interface is in standby mode, ready to begin forwarding traffic if the primary link shuts down. The interface being configured is referred to as the active link; the specified interface is identified as the backup link. The feature provides an alternative to the Spanning Tree Protocol (STP), allowing users to turn off STP and still retain basic link redundancy.

This command is available only for Layer 2 interfaces.

You can configure only one Flex Link backup link for any active link, and it must be a different interface from the active interface.

- An interface can belong to only one Flex Link pair. An interface can be a backup link for only one active link. An active link cannot belong to another Flex Link pair.
- A backup link does not have to be the same type (Fast Ethernet or Gigabit Ethernet, for instance) as the active link. However, you should configure both Flex Links with similar characteristics so that there are no loops or changes in behavior if the standby link begins to forward traffic.
- Neither of the links can be a port that belongs to an EtherChannel. However, you can configure two port channels (EtherChannel logical interfaces) as Flex Links, and you can configure a port channel and a physical interface as Flex Links, with either the port channel or the physical interface as the active link.
- If STP is configured on the switch, Flex Links do not participate in STP in all valid VLANs. If STP is not running, be sure that there are no loops in the configured topology.

This example shows how to configure two interfaces as Flex Links:

```
Device# configure terminal
Device(conf)# interface gigabitethernet1/0/1
Device(conf-if)# switchport backup interface gigabitethernet1/0/2
Device(conf-if)# end
```

This example shows how to configure the Gigabit Ethernet interface to always preempt the backup:

```
Device# configure terminal
Device(conf)# interface gigabitethernet1/0/1
Device(conf-if)# switchport backup interface gigabitethernet1/0/2 preemption forced
Device(conf-if)# end
```

This example shows how to configure the Gigabit Ethernet interface preemption delay time:

```
Device# configure terminal
Device(conf)# interface gigabitethernet1/0/1
Device(conf-if)# switchport backup interface gigabitethernet1/0/2 preemption delay 150
Device(conf-if)# end
```

This example shows how to configure the Gigabit Ethernet interface as the MMU primary VLAN:

```
Device# configure terminal
Device(conf)# interface gigabitethernet1/0/1
Device(conf-if)# switchport backup interface gigabitethernet1/0/2 mmu primary vlan 1021
Device(conf-if)# end
```

You can verify your setting by entering the **show interfaces switchport backup** privileged EXEC command.



# switchport block

To prevent unknown multicast or unicast packets from being forwarded, use the **switchport block** command in interface configuration mode. To allow forwarding unknown multicast or unicast packets, use the **no** form of this command.

```
switchport block {multicast | unicast}
no switchport block {multicast | unicast}
```

<b>Syntax Description</b>	<p><b>multicast</b> Specifies that unknown multicast traffic should be blocked.</p> <p><b>Note</b> Only pure Layer 2 multicast traffic is blocked. Multicast packets that contain IPv4 or IPv6 information in the header are not blocked.</p> <p><b>unicast</b> Specifies that unknown unicast traffic should be blocked.</p>				
<b>Command Default</b>	Unknown multicast and unicast traffic is not blocked.				
<b>Command Modes</b>	Interface configuration				
<b>Command History</b>	<table border="1"> <thead> <tr> <th>Release</th> <th>Modification</th> </tr> </thead> <tbody> <tr> <td>Cisco IOS Release 15.0(2)EX</td> <td>This command was introduced.</td> </tr> </tbody> </table>	Release	Modification	Cisco IOS Release 15.0(2)EX	This command was introduced.
Release	Modification				
Cisco IOS Release 15.0(2)EX	This command was introduced.				
<b>Usage Guidelines</b>	<p>By default, all traffic with unknown MAC addresses is sent to all ports. You can block unknown multicast or unicast traffic on protected or nonprotected ports. If unknown multicast or unicast traffic is not blocked on a protected port, there could be security issues.</p> <p>With multicast traffic, the port blocking feature blocks only pure Layer 2 packets. Multicast packets that contain IPv4 or IPv6 information in the header are not blocked.</p> <p>Blocking unknown multicast or unicast traffic is not automatically enabled on protected ports; you must explicitly configure it.</p> <p>For more information about blocking packets, see the software configuration guide for this release.</p> <p>This example shows how to block unknown unicast traffic on an interface:</p> <pre>Device(config-if)# switchport block unicast</pre> <p>You can verify your setting by entering the <b>show interfaces interface-id switchport</b> privileged EXEC command.</p>				

# system mtu

**system mtu** {*bytes* | **jumbo** *bytes*}

**no system mtu**

<b>Syntax Description</b>	<i>bytes</i>	Set the system MTU for ports that are set to 10 or 100 Mb/s. The range is 1500 to 1998 bytes. This is the maximum MTU received at 10/100-Mb/s Ethernet switch ports.
	<b>jumbo</b> <i>bytes</i>	Set the system jumbo MTU for Gigabit Ethernet ports operating at 1000 Mb/s or greater. The range is 1500 to 9000 bytes. This is the maximum MTU received at the physical port for Gigabit Ethernet ports.

**Command Default** The default MTU size for all ports is 1500 bytes.

**Command Modes** Global configuration

<b>Command History</b>	<b>Release</b>	<b>Modification</b>
	Cisco IOS Release 15.0(2)EX	This command was introduced.

**Usage Guidelines** The switch does not support the MTU on a per-interface basis.

When you use this command to change the system MTU or jumbo MTU size, you must reset the switch before the new configuration takes effect. The system MTU setting is saved in the switch environmental variable in NVRAM and becomes effective when the switch reloads. The MTU settings you enter with the **system mtu** and **system mtu jumbo** commands are not saved in the switch IOS configuration file, even if you enter the **copy running-config startup-config** privileged EXEC command. Therefore, if you use TFTP to configure a new switch by using a backup configuration file and want the system MTU to be other than the default, you must explicitly configure the **system mtu** and **system mtu jumbo** settings on the new switch and then reload the switch.

Gigabit Ethernet ports operating at 1000 Mb/s are not affected by the **system mtu** command, and 10/100-Mb/s ports are not affected by the **system mtu jumbo** command.

If you enter a value that is outside the range for the specific type of switch, the value is not accepted.

You can verify your setting by entering the **show system mtu** privileged EXEC command.

This example shows how to set the global system MTU size to 1600 bytes:

```
Device(config)# system mtu 1600
Changes to the system MTU will not take effect until the next reload is done
```

```
Device(config)#
```

This example shows how to set the global system MTU size to 6000 bytes:

```
Device(config)# system mtu jumbo 6000
Changes to the system jumbo MTU will not take effect until the next reload is done
```

```
Device(config)#
```

## voice-signaling vlan (network-policy configuration)

To create a network-policy profile for the voice-signaling application type, use the **voice-signaling vlan** command in network-policy configuration mode. To delete the policy, use the **no** form of this command.

```
voice-signaling vlan {vlan-id [{cos cos-value | dscp dscp-value}] | dot1p [{cos l2-priority | dscp dscp}] | none | untagged}
```

Syntax Description	
<b>vlan-id</b>	(Optional) The VLAN for voice traffic. The range is 1 to 4094.
<b>cos</b> <i>cos-value</i>	(Optional) Specifies the Layer 2 priority class of service (CoS) for the configured VLAN. The range is 0 to 7; the default is 5.
<b>dscp</b> <i>dscp-value</i>	(Optional) Specifies the differentiated services code point (DSCP) value for the configured VLAN. The range is 0 to 63; the default is 46.
<b>dot1p</b>	(Optional) Configures the phone to use IEEE 802.1p priority tagging and to use VLAN 0 (the native VLAN).
<b>none</b>	(Optional) Does not instruct the Cisco IP phone about the voice VLAN. The phone uses the configuration from the phone key pad.
<b>untagged</b>	(Optional) Configures the phone to send untagged voice traffic. This is the default for the phone.

**Command Default** No network-policy profiles for the voice-signaling application type are defined.  
 The default CoS value is 5.  
 The default DSCP value is 46.  
 The default tagging mode is untagged.

**Command Modes** Network-policy profile configuration

Command History	Release	Modification
	Cisco IOS Release 15.0(2)EX	This command was introduced.

**Usage Guidelines** Use the **network-policy profile** global configuration command to create a profile and to enter network-policy profile configuration mode.

The voice-signaling application type is for network topologies that require a different policy for voice signaling than for voice media. This application type should not be advertised if all of the same network policies apply as those advertised in the voice policy TLV.

When you are in network-policy profile configuration mode, you can create the profile for voice-signaling by specifying the values for VLAN, class of service (CoS), differentiated services code point (DSCP), and tagging mode.

These profile attributes are contained in the Link Layer Discovery Protocol for Media Endpoint Devices (LLDP-MED) network-policy time-length-value (TLV).

To return to privileged EXEC mode from the network-policy profile configuration mode, enter the **exit** command.

This example shows how to configure voice-signaling for VLAN 200 with a priority 2 CoS:

```
Device(config)# network-policy profile 1  
Device(config-network-policy)# voice-signaling vlan 200 cos 2
```

This example shows how to configure voice-signaling for VLAN 400 with a DSCP value of 45:

```
Device(config)# network-policy profile 1  
Device(config-network-policy)# voice-signaling vlan 400 dscp 45
```

This example shows how to configure voice-signaling for the native VLAN with priority tagging:

```
Device(config-network-policy)# voice-signaling vlan dot1p cos 4
```

## voice vlan (network-policy configuration)

To create a network-policy profile for the voice application type, use the **voice vlan** command in network-policy configuration mode. To delete the policy, use the **no** form of this command.

```
voice vlan {vlan-id [{cos cos-value | dscp dscp-value}] | dot1p [{cos l2-priority | dscp dscp}] | none | untagged}
```

Syntax Description		
<b>vlan-id</b>	(Optional)	The VLAN for voice traffic. The range is 1 to 4094.
<b>cos</b> <i>cos-value</i>	(Optional)	Specifies the Layer 2 priority class of service (CoS) for the configured VLAN. The range is 0 to 7; the default is 5.
<b>dscp</b> <i>dscp-value</i>	(Optional)	Specifies the differentiated services code point (DSCP) value for the configured VLAN. The range is 0 to 63; the default is 46.
<b>dot1p</b>	(Optional)	Configures the phone to use IEEE 802.1p priority tagging and to use VLAN 0 (the native VLAN).
<b>none</b>	(Optional)	Does not instruct the Cisco IP phone about the voice VLAN. The phone uses the configuration from the phone key pad.
<b>untagged</b>	(Optional)	Configures the phone to send untagged voice traffic. This is the default for the phone.

**Command Default** No network-policy profiles for the voice application type are defined.  
The default CoS value is 5.  
The default DSCP value is 46.  
The default tagging mode is untagged.

**Command Modes** Network-policy profile configuration

Command History	Release	Modification
	Cisco IOS Release 15.0(2)EX	This command was introduced.

**Usage Guidelines** Use the **network-policy profile** global configuration command to create a profile and to enter network-policy profile configuration mode.

The voice application type is for dedicated IP telephones and similar devices that support interactive voice services. These devices are typically deployed on a separate VLAN for ease of deployment and enhanced security through isolation from data applications.

When you are in network-policy profile configuration mode, you can create the profile for voice by specifying the values for VLAN, class of service (CoS), differentiated services code point (DSCP), and tagging mode.

These profile attributes are contained in the Link Layer Discovery Protocol for Media Endpoint Devices (LLDP-MED) network-policy time-length-value (TLV).

To return to privileged EXEC mode from the network-policy profile configuration mode, enter the **exit** command.

This example shows how to configure the voice application type for VLAN 100 with a priority 4 CoS:

```
Device(config)# network-policy profile 1  
Device(config-network-policy)# voice vlan 100 cos 4
```

This example shows how to configure the voice application type for VLAN 100 with a DSCP value of 34:

```
Device(config)# network-policy profile 1  
Device(config-network-policy)# voice vlan 100 dscp 34
```

This example shows how to configure the voice application type for the native VLAN with priority tagging:

```
Device(config-network-policy)# voice vlan dot1p cos 4
```